



TUGAS AKHIR -TE 141599

SOLUSI *INVERSE KINEMATICS* MENGGUNAKAN LOGIKA *FUZZY* PADA ROBOT MANIPULATOR DENSO 6 DoF

Septian Surya Pradana
NRP 2212 106 051

Dosen Pembimbing
Dr. Trihastuti Agustinah, S.T., M.T.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2015



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT-TE 141599

INVERSE KINEMATICS SOLUTION BASED ON FUZZY LOGIC FOR THE DENSO 6 DOF ROBOT MANIPULATOR

Septian Surya Pradana
NRP 2212 106 051

Supervisor
Dr. Trihastuti Agustinah, S.T., M.T.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2015

**SOLUSI INVERSE KINEMATICS MENGGUNAKAN LOGIKA
FUZZY PADA ROBOT MANIPULATOR DENSO 6-DOF**

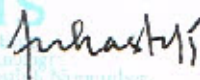
TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada**

**Bidang Studi Teknik Sistem Pengaturan
Jurusan Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing,



Dr. Trihastuti Agustinah, S.T., M.T.
NIP: 196808121994032001



SOLUSI *INVERSE KINEMATICS* MENGGUNAKAN LOGIKA *FUZZY* PADA ROBOT MANIPULATOR DENSO 6 DoF

Nama : Septian Surya Pradana
Pembimbing : Dr. Trihastuti Agustinah, S.T., M.T.

ABSTRAK

Dalam perhitungan nilai *theta* setiap sendi pada robot manipulator diperlukan metode perancangan *inverse kinematics*, dalam perhitungan ini dibutuhkan masukan untuk posisi akhir robot. Nilai yang dibutuhkan adalah titik koordinat translasi x , y dan z serta posisi orientasi robot terhadap sumbu x , y dan z . Metode yang digunakan untuk perhitungan *inverse kinematics* adalah metode aljabar. Algoritma *inverse kinematics* yang efisien tidak membutuhkan nilai *inverse* dari setiap matriks yang didapat dari setiap nilai *link* pada DH-Parameter untuk robot manipulator Denso 6-DoF. Tahap-tahap yang dibutuhkan untuk mencari nilai *inverse kinematics* adalah pertama menemukan nilai *theta* 1, kemudian *theta* 2, *theta* 3, *theta* 5, *theta* 4 dan yang terakhir *theta* 6.

Pada Tugas Akhir ini, perhitungan untuk mencari nilai *theta* 1 menggunakan sistem *fuzzy*. Perancangan sistem *fuzzy* didasarkan pada beberapa nilai yang didapat dari perhitungan manual untuk *theta* 1. Dari hasil beberapa simulasi menggunakan perbandingan antara hasil sistem *fuzzy* dengan perhitungan manual menghasilkan rata-rata error sebesar 4.761.

Kata Kunci : *Inverse Kinematics*, Sistem *Fuzzy*, Sugeno

--- *Halaman ini sengaja dikosongkan* ---

INVERSE KINEMATICS SOLUTION BASED ON FUZZY LOGIC FOR THE DENSO 6 DOF ROBOT MANIPULATOR

Name : Septian Surya Pradana
Supervisor : Dr. Trihastuti Agustinah, S.T., M.T.

ABSTRACT

Calculation of theta value on every robot manipulator needs inverse kinematics design method, for the calculation needs input for last position of robot. The required value is translation coordinate point x, y and z also robot orientation point to x, y and z axis. Method for inverse kinematic's calculation is algebra method. Efficient algorithm of inverse kinematics calculation does not need inverse value of every matrices which obtain from every link value on the DH-Parameter for Denso 6-DoF robot manipulator. Require steps to find inverse kinematics value is firstly find theta 1 value, then theta 2, theta 3, theta 5, theta 4 and the last is theta 6.

On this Final Project, calculation to find theta 1 use fuzzy system. Fuzzy system reference design is based on some values that find from manual calculation for theta 1. From some simulation results use comparison between fuzzy system result and manual calculation deliver a result of total error average as big as 4.761.

Keyword: Inverse Kinematics, Fuzzy System, Sugeno

--- *Halaman ini sengaja dikosongkan* ---

KATA PENGANTAR

Segala puji dan syukur penulis panjatkan kehadirat Allah SWT karena atas segala rahmat, taufik dan hidayah-Nya sehingga Tugas Akhir ini dapat terselesaikan dengan baik. Tugas Akhir dengan judul:

SOLUSI *INVERSE KINEMATICS* MENGGUNAKAN LOGIKA *FUZZY* PADA ROBOT MANIPULATOR DENSO 6 DoF

dibuat untuk memenuhi persyaratan guna menyelesaikan pendidikan Strata-1 pada Bidang Studi Teknik Sistem Pengaturan, Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya.

Pada kesempatan ini penulis menyampaikan terima kasih atas segala bantuan dan dukungannya yang telah diberikan selama proses pembuatan Tugas Akhir ini kepada :

1. Kedua orang tua atas dukungan baik spiritual maupun material yang tak ternilai harganya.
2. Ibu Dr. Trihastuti Agustinah, ST., MT., selaku dosen yang membimbing dan memberikan sebagian ilmunya pada penulis. Bapak dan Ibu Dosen Teknik Sistem Pengaturan.
3. Seluruh teman-teman dari Teknik Sistem Pengaturan, antara lain Syfa, Nani, Duli, Ikhsan E, Ihsan R, Anas, Bustanudin, Olenka, Anna, Yuda, Aji, Jahlun, Gunarko, Yusuf, Bagas, Pras, Wildan dan Trio yang telah memberikan dukungan dan semangat mengerjakan Tugas Akhir.
4. Fahmi Al-Ghifari, yang telah membantu dalam pembuatan desain SOLIDWORK.
5. Semua pihak yang telah membantu kami dalam penyelesaian Tugas Akhir ini yang tidak dapat kami sebutkan satu persatu.

Penulis sadar bahwa masih banyak kekurangan dalam penelitian maupun dalam penulisan buku ini. Untuk itu, Penulis mohon maaf atas segala kekurangan tersebut. Semoga Tugas Akhir ini dapat bermanfaat untuk pengembangan ilmu pengetahuan dan teknologi di kemudian hari.

Surabaya, Januari 2015

Penulis

--- *Halaman ini sengaja dikosongkan* ---

DAFTAR ISI

	Halaman
JUDUL	i
JUDUL (ENGLISH)	iii
PERNYATAAN KEASLIAN	v
LEMBAR PENGESAHAN	vii
ABSTRAK	ix
ABSTRAK (ENGLISH)	xi
KATA PENGANTAR	xiii
DAFTAR ISI	xv
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
 BAB I PENDAHULUAN	 1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian	2
1.5 Metodologi Penelitian	2
1.6 Sistematika Penulisan.....	3
1.7 Relevansi	4
 BAB II TEORI DASAR	 5
2.1 Tinjauan Pustaka	5
2.2 Robot manipulator DENSO 6-DoF	6
2.3 Pemodelan Robot	7
2.3.1 Pembentukan Sendi	7
2.3.2 Transformasi Homogen	9
2.4 Kinematika	11
2.4.1 Kinematika Maju (Forward Kinemaitcs).....	11
2.4.1.1 DH-Parameters	11
2.4.1.2 Penghitungan kinematika maju	12
2.4.2 Kinematika Terbalik (<i>Inverse Kinematics</i>).....	13
2.4.2.1 Pendekatan Geometri	13
2.4.2.2 Pendekatan Aljabar.....	14
2.4.2.3 Menghitung theta.....	15
2.5Sistem <i>Fuzzy</i>	16
2.5.1 Himpunan <i>Fuzzy</i>	17
2.5.2 Fungsi Keanggotaan <i>Fuzzy</i>	17

2.5.2.1 Fungsi Keanggotaan Segitiga	17
2.5.2.2 Fungsi Keanggotaan Trapesium	18
2.5.2.3 Fungsi Keanggotaan Gaussian	19
2.5.3 Operasi Himpunan <i>Fuzzy</i>	19
2.5.4 Sistem Inferensi <i>Fuzzy</i>	20
2.5.5 Metode Sugeno.....	22
BAB III DESAIN SISTEM	23
3.1 Pemodelan Transformasi Homogen Denso 6-DoF.....	23
3.1.1 DH-Parameter Robot Denso 6-DoF	23
3.1.2 Model Transformasi Homogen tiap <i>Link</i>	23
3.2 Pemodelan Robot Denso 6-DOF pada SOLIDWORK.....	25
3.2.1 Rancangan <i>Base Frame</i> dan setiap <i>Link</i>	25
3.2.2 Rancangan keseluruhan Robot	27
3.3 Kinematics	28
3.3.1 Kinematika Maju (<i>Forward Kinematics</i>)	28
3.3.2 Kinematika Terbalik (<i>Inverse Kinematics</i>).....	30
3.4 Desain Sistem <i>Fuzzy</i>	32
3.5 Diagram blok keseluruhan	35
BAB IV HASIL DAN SIMULASI.....	37
4.1 Pengujian Pemodelan <i>Fuzzy</i> dengan Masukan.....	37
4.2 Pengujian Model Rumus <i>Inverse Kinematics</i>	42
4.3 Perbandingan hasil solusi	44
BAB V PENUTUP	49
5.1 Kesimpulan	49
5.2 Saran	49
DAFTAR PUSTAKA	51
LAMPIRAN	53
RIWAYAT PENULIS	67

DAFTAR GAMBAR

Gambar 2.1	Model dari robot manipulator Denso 6-DoF	6
Gambar 2.2	Beberapa variasi bentuk sendi <i>revolute</i>	7
Gambar 2.3	Bentuk sendi <i>prismatic</i>	8
Gambar 2.4	Bentuk sumbu sebagai referensi sendi	8
Gambar 2.5	Bentuk pemodelan tiap sendi robot Denso 6-DoF.....	8
Gambar 2.6	Perbedaan antara kinematika maju dan terbalik	11
Gambar 2.7	Pengaplikasian pendekatan Geometri	13
Gambar 2.8	Fungsi <i>Atan2</i> pada empat kuadran	15
Gambar 2.9	Tata letak <i>Fuzzy Logic Toolbox</i> pada MATLAB	17
Gambar 2.10	Fungsi Keanggotaan Segitiga	18
Gambar 2.11	Fungsi Keanggotaan Trapesium.....	19
Gambar 2.12	Fungsi Keanggotaan <i>Gaussian</i>	19
Gambar 2.13	Sistem Inferensi <i>Fuzzy</i>	20
Gambar 2.14	<i>FIS Editor</i> pada <i>Toolbox Fuzzy</i> MATLAB	22
Gambar 3.1	Panjang setiap <i>link</i> dalam satuan milimeter	25
Gambar 3.2	Model <i>Base Frame</i> atau <i>Link 1</i>	25
Gambar 3.3	Model <i>Link 2</i>	26
Gambar 3.4	Model <i>Link 3</i>	26
Gambar 3.5	Model <i>Link 4</i>	26
Gambar 3.6	Model <i>Link 5</i>	27
Gambar 3.7	Model <i>Link 6</i>	27
Gambar 3.8	Model <i>End-effector</i>	27
Gambar 3.9	Pemodelan robot Denso 6-DoF	28
Gambar 3.10	Fungsi keanggotaan nilai pX	34
Gambar 3.11	Fungsi keanggotaan nilai pY	34
Gambar 3.12	Diagram blok keseluruhan.....	36
Gambar 4.1	Grafik Variasi pertama θ_1	38
Gambar 4.2	Grafik Variasi pertama θ_2	40
Gambar 4.3	Grafik Variasi pertama θ_3	41
Gambar 4.4	Beberapa Gerakan Robot Solusi Kinematika Terbalik	44
Gambar 4.5	Grafik Perbandingan Pertama.....	45
Gambar 4.6	Gerakan Robot Perbandingan Pertama.....	45
Gambar 4.7	Grafik Perbandingan Kedua	46
Gambar 4.8	Gerakan Robot Perbandingan Kedua	47
Gambar 4.9	Grafik Perbandingan Ketiga	48
Gambar 4.10	Gerakan Robot Perbandingan Ketiga.....	48

--- *Halaman ini sengaja dikosongkan* ---

DAFTAR TABEL

Tabel 2.1 Beberapa persamaan trigonometri kinematika terbalik	16
Tabel 3.1 DH-Parameter robot Denso 6-DOF.....	23
Tabel 3.2 Beberapa Solusi Kemungkinan θ_1	33
Tabel 4.1 Hasil Variasi hitungan pertama.....	37
Tabel 4.2 Hasil Variasi hitungan kedua	39
Tabel 4.3 Hasil Variasi hitungan ketiga.....	40
Tabel 4.4 Beberapa hasil solusi kinematika terbalik.....	42
Tabel 4.5 Hasil Perbandingan pertama..	44
Tabel 4.6 Hasil Perbandingan kedua.....	46
Tabel 4.7 Hasil Perbandingan ketiga.....	47

--- *Halaman ini sengaja dikosongkan* ---

BAB I

PENDAHULUAN

Bab ini menjelaskan latar belakang, perumusan masalah, dan tujuan dari penelitian pada Tugas Akhir ini. Selain itu, dijelaskan juga mengenai metodologi dan sistematika dalam penyelesaian Tugas Akhir, serta relevansi untuk penelitian selanjutnya.

1.1 Latar Belakang

Robotika ialah ilmu yang mempelajari tentang robot yang meliputi beberapadisiplin ilmu (elektronika, mekanik, kontrol, komputer, dll). Istilah robot berawal bahasa Ceko “robota” yang berarti pekerja atau kuli yang tidak mengenal lelah atau bosan. Robot biasanya digunakan untuk tugas yang berat, berbahaya, pekerjaan yang berulang dan kotor. Biasanya kebanyakan robot industri digunakan dalam bidang produksi. Sistem koordinat pada Robotika memiliki 3 derajat kebebasan (*Degree of Freedom*) atau 3 axis, yang terdiri dari (θ) mewakili sumbu putar, sumbu z mewakili gerakan naik turun serta sumbu R yang mewakili gerakan memanjang atau memendek.

Spesifikasi lintasan yang diinginkan dari robot manipulator biasanya diberikan pada ruang koordinat (*Cartesian*). Sistem dirancang dalam ruang sendi, oleh karena itu diperlukan proses pemetaan antara dua ruang tersebut. Pemetaan untuk robot manipulator umumnya diketahui sebagai kinematika maju (*forward kinematics*) yaitu kondisi ketika ruang sendi dikonversi ke ruang koordinat dan kinematika terbalik (*inverse kinematics*) kondisi ketika ruang koordinat dikonversi ke ruang sendi.

Beberapa peneliti telah melakukan perhitungan untuk mencari nilai θ pada kinematika terbalik [1]. Perhitungan dilakukan dengan mengacu pada pendekatan aljabar, sehingga perhitungan menjadi lebih efisien yaitu dengan mentransformasikan persamaan matriks menjadi 8 persamaan yang mengandung 6 sudut yang tidak diketahui. Dalam tahap pemetaan kinematika terdapat beberapa kesulitan yaitu intensif kompleksitas komputasi dari efek *coupling* robot dan kesulitan untuk memperoleh solusi bentuk tertutup dari kinematika terbalik. Penyelesaian masalah untuk kinematika terbalik lebih kompleks dibandingkan dengan kinematika maju untuk robot dengan struktur seri

dan paralel. Hal ini dikarenakan penyelesaian dari persamaan kinematika terbalik lebih kompleks dan nonlinear.

Pada Tugas Akhir ini, akan dilakukan perhitungan nilai *theta* dengan menggunakan sistem *fuzzy* untuk memperoleh solusi sistem nonlinear yang kompleks. Perhitungan *theta* 1 untuk kinematika terbalik robot manipulator akan diperoleh menggunakan sistem *fuzzy*.

1.2 Rumusan Masalah

Permasalahan yang akan diselesaikan pada Tugas Akhir ini yaitu desain sistem *fuzzy* untuk pencarian nilai *theta* sendi pertama yang akan menjadi referensi bagi persamaan selanjutnya untuk mencari nilai *theta* yang lain. Nilai seluruh *theta* akan menentukan pergerakan robot berdasarkan dari masukan nilai translasi titik *x*, *y* dan *z* dengan menggunakan persamaan *inverse kinematics* untuk seluruh nilai *theta*.

1.3 Batasan Masalah

Perhitungan dibatasi tidak sampai membahas *singularity*. Titik *singularity* menyebabkan nilai *theta* pada tiap sendi tidak ditemukan sehingga robot tidak dapat berjalan. Perhitungan dibatasi sampai kinematika terbalik dan tidak mencakup perhitungan dinamika robot (*dynamic*).

1.4 Tujuan Penelitian

Penelitian pada Tugas Akhir ini bertujuan untuk mencari nilai *theta* pada robot manipulator Denso 6-DoF. Dengan melakukan perhitungan kinematika terbalik maka akan diperoleh nilai *theta* sehingga pergerakan robot dapat ditentukan sesuai dengan posisi akhir yang diinginkan. Untuk mencapai posisi akhir yang diinginkan, terdapat beberapa solusi pergerakan dari hasil perhitungan kinematika terbalik. Dari beberapa solusi tersebut akan dipilih 1 solusi pergerakan, kemudian dilakukan perancangan sistem *fuzzy* untuk mendapat nilai *theta* pertama pada robot. Konsep *fuzzy* yang digunakan pada tahapan perancangan sistem adalah model *fuzzy* tipe Sugeno.

1.5 Metodologi Penelitian

Metodologi yang digunakan dalam penyusunan Tugas Akhir ini, antara lain:

1. Studi Literatur
Pada tahap ini dilakukan kajian teori mengenai perhitungan kinematika terbalik untuk robot manipulator 6-DoF, desain robot virtual dan desain sistem *fuzzy* untuk robot 6-DoF.
2. Pemodelan Sistem
Pencarian tiap sudut dari robot 6-DoF diperoleh dengan menggunakan pendekatan aljabar. Hasil dari pencarian sudut tersebut akan dijadikan referensi untuk tahap perancangan sistem *fuzzy*.
3. Desain sistem *Fuzzy*
Pada tahap ini dilakukan desain sistem *fuzzy* berupa nilai-nilai *theta* yang didapat dari tahap pemodelan sistem, sehingga akan diperoleh nilai *theta* 1 pada tiap solusi sesuai dengan nilai akhir yang diinginkan.
4. Simulasi
Hasil desain sistem *fuzzy* selanjutnya disimulasikan dengan bantuan *software* MATLAB R2012a dengan menggabungkan desain robot yang telah dibentuk pada SOLIDWORK 2014.
5. Penulisan Buku
Tahap penyusunan buku, meliputi pendahuluan, teori dasar, desain sistem, hasil dan simulasi serta kesimpulan sebagai penutup.

1.6 Sistematika Penulisan

Pembahasan pada Tugas Akhir ini dibagi menjadi lima bab dengan sistematika penulisan sebagai berikut:

BAB I : PENDAHULUAN

Pada bab ini dijelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, sistematika penulisan, serta relevansi.

BAB II : TEORI DASAR

Bab ini menjelaskan tentang tinjauan pustaka, pemodelan robot manipulator Denso 6-DoF, kinematika dan teori sistem *fuzzy*.

BAB III : DESAIN SISTEM

Bagian ini berisi pembahasan mengenai perumusan untuk setiap sendi robot manipulator Denso 6- DoF serta perancangan sistem *fuzzy* berdasarkan teori pada Bab II.

BAB IV : HASIL DAN SIMULASI

Bab ini memuat hasil simulasi beserta analisis dari tiap data pengujian.

BAB V : PENUTUP

Analisis yang dilakukan pada Bab IV akan mendapatkan suatu kesimpulan. Saran diberikan sebagai bahan evaluasi untuk penelitian selanjutnya.

1.7 Relevansi

Hasil akhir yang diperoleh dari penelitian pada Tugas Akhir ini dapat dijadikan sebagai referensi untuk mempelajari persamaan untuk kinematika terbalik robot 3-DoF atau lebih serta pengembangan sistem *fuzzy* yang tepat untuk menemukan sudut pada kinematika terbalik.

BAB II TEORI DASAR

Pada Bab II berisi materi dasar yang digunakan dalam penyelesaian dari Tugas Akhir ini. Beberapa aspek yang akan dibahas pada Bab ini meliputi tinjauan pustaka terkait penelitian robot manipulator Denso 6-DoF, *modeling* robot manipulator Denso 6-DoF, teori kinematika serta teori mengenai sistem *fuzzy*.

2.1 Tinjauan Pustaka

Penelitian mengenai pemodelan kinematika terbalik (*inverse kinematic*) pada robot PUMA560 menggunakan algoritma yang efisien berdasarkan pendekatan aljabar [1]. Dengan memanfaatkan pemodelan kinematik berdasar DH-parameter yang telah ditentukan akan dirancang perhitungan yang hasilnya lebih efisien karena tidak menghitung *invers* dari masing – masing transformasi homogen setiap *link*. Nilai derajat setiap sendi (*joint theta*) dari *link* didapat memanfaatkan eliminasi dari persamaan yang didapat pada tahap sebelumnya.

Berdasarkan derajat yang diperoleh untuk setiap sendi (*joint*) pada tahap ini dimanfaatkan untuk beberapa aplikasi dan perhitungan *dynamic* dari robot 6-DoF. Berdasar ilmu kinematika, kinematik adalah ilmu yang mempelajari pergerakan robot dengan mengabaikan gaya– gaya yang terjadi.

Telah dilakukan penelitian mengenai solusi kinematika terbalik dari robot lengan 7-DOF dengan menggunakan sistem *fuzzy* [2]. Dari perhitungan manual didapatkan perhitungan untuk derajat dari setiap *link*. Setelah didapatkan nilai dari perhitungan manual, penyusunan sistem *fuzzy* untuk mencari nilai derajat sendi pertama berdasarkan nilai untuk titik – titik x dan z [2]. Perolehan derajat sendi yang lainnya tetap mengambil dari perhitungan rumus yang didapat.

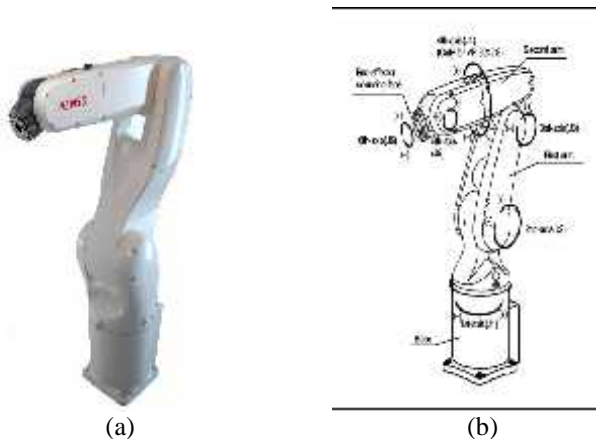
Serupa dengan penelitian [1], pada Tugas Akhir ini dihitung untuk mencari rumus masing – masing derajat sendi dengan pendekatan aljabar pada robot manipulator Denso 6-DoF. Setelah didapatkannya rumus perhitungan derajat maka akan dirancang sistem *fuzzy* untuk mencari derajat sendi dari *link* pertama seperti pada penelitian [2], yang menjadi acuan Tugas Akhir karena perancangan sistem *fuzzy* digunakan untuk derajat sendi pertama. Desain untuk simulasi menggunakan

software *simulink* MATLAB 2012 dengan mengintegrasikan CAD model pada software SOLIDWORK 2014 menggunakan SimMechanics.

2.2 Robot manipulator DENSO 6-DoF

Bentuk keseluruhan dari robot manipulator Denso 6-DoF terlihat pada Gambar 2.1 (a), dan kerangka gerakan dari robot manipulator dapat dilihat pada Gambar 2.1 (b) dengan memperlihatkan bahwa keseluruhan sendi pada robot tersebut merupakan sendi *revolute* yang berarti gerakan dari setiap sendi hanya bisa rotasi saja. Dari gambar tersebut dapat dikategorikan bahwa robot ini mempunyai struktur paralel. Pengaksesan secara langsung pada robot ini menggunakan model *Simulink* pada MATLAB, bagian *end-effector* pada robot ini dapat dikombinasikan dengan beberapa peralatan yang diinginkan, misalkan robot akan digunakan untuk menggambar maka bagian *end-effector* dapat ditambahi bor dan sebagainya.

Prinsip gerakan robot 6-DoF dibagi menjadi dua, yaitu gerakan untuk menentukan posisi robot dan gerakan pergelangan robot. Posisi robot ditentukan berdasarkan gerakan sendi 1 sampai 3, sedangkan gerakan luar pergelangan tangan didasarkan pada gerakan sendi 4 sampai 6. Pergerakan dari sendi-sendi tersebut mempengaruhi ruang kerja dari robot tersebut.



Gambar 2.1 (a) Model dari robot manipulator Denso 6-DoF
(b) Kerangka Pergerakan robot manipulator Denso 6-DoF

Perancangan desain untuk pergerakan robot membutuhkan sistem mekanik seperti hidrolik dan elektrik. Pada robot Denso 6-DoF memiliki tuas pengendali / *Haptic Device* yang memungkinkan untuk menggerakkan tiap sendi pada robot yang telah diintegrasikan sebelumnya dengan *Denso Controller*.

Desain pemodelan pada robot manipulator dirancang berdasarkan DH-Parameter yang terbentuk dari robot. Pemodelan antar link robot dapat terintegrasikan dan dihitung pergerakan rotasi. Hasil pemodelan akan disimulasikan menggunakan Solidwork CAD model dan digerakkan menggunakan MATLAB R2012a sesuai perhitungan yang didapat. Penjelasan mengenai desain sistem secara keseluruhan akan dijelaskan pada Bab 3.

2.3 Pemodelan Robot

Pengetahuan mengenai berbagai macam sendi dan pemodelan matriks Transformasi Homogen merupakan hal terpenting dalam mengetahui pemodelan suatu robot. Pemodelan robot adalah hal terpenting untuk dapat menghitung kinematika maju dan kinematika terbalik.

2.3.1 Pembentukan Sendi

Sendi pada robot berfungsi untuk menghubungkan dua *link*, sehingga antara *link* tersebut dapat menentukan arah putaran atau gerakan yang diinginkan sesuai dengan bentuk sendinya. Sendi pada robot dapat didefinisikan menjadi dua bentuk, yaitu:

1. Sendi *revolute*

Sendi *revolute* adalah sendi yang bergerak secara berputar / rotasi.

Bentuk umum dari sendi *revolute* terlihat pada Gambar 2.2.

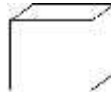


Gambar 2.2 Beberapa variasi bentuk sendi *revolute*

2. Sendi *prismatic*

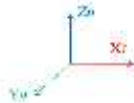
Sendi *prismatic* merupakan sendi yang bergerak secara translasi.

Bentuk umum dari sendi *prismatic* terlihat pada Gambar 2.3.



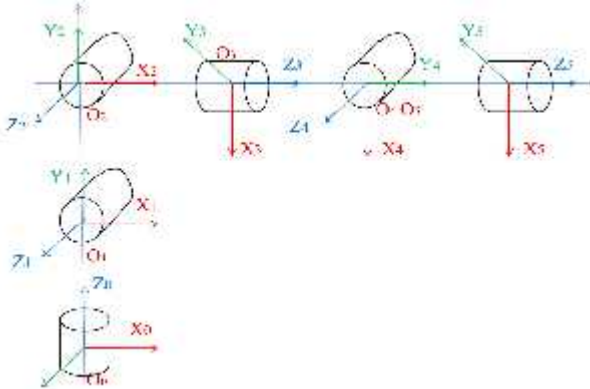
Gambar 2.3 Bentuk sendi *prismatic*

Sedangkan sumbu untuk menentukan koordinat robot menjadi referensi dari sendi untuk melakukan pergerakan dan pembentukan *link* robot. Gambar sumbu dapat terlihat pada Gambar 2.4. Penentuan koordinat yang dibentuk dari sumbu ini didapatkan dari koordinat pada *software* SOLIDWORK 2014.



Gambar 2.4 Bentuk sumbu sebagai referensi sendi

Bentuk keseluruhan sendi lengkap dengan koneksi seluruh link pada robot manipulator Denso 6-DoF diilustrasikan pada Gambar 2.5.



Gambar 2.5 Bentuk pemodelan tiap sendi robot Denso 6-DoF

Dalam pembentukan koneksi keseluruhan sendi tersebut dibutuhkan beberapa langkah, yaitu

- Meletakkan dan memberi Label Z_i , ... Z_n yang lurus terhadap sumbu originnya.
- Menentukan Base Frame, Letakkan X_0 , Y_0 dan Z_0 sesuai kaidah

tangan kanan.

- c. Meletakkan O_i yang common normal terhadap Z_i dan Z_{i-1} letakkan O_i pada perpotongan. Jika Z_i dan Z_{i-1} parallel letakkan O_i pada sendi ke i .
- d. Menentukan X_i yang common normal antara Z_i dan Z_{i-1} yang melalui O_i . Atau direction normal jika Z_i dan Z_{i-1} berpotongan.
- e. Menentukan Y_i sesuai kaidah tangan kanan.
- f. Menentukan *end-effector*, sesuaikan sumbu $X_n Y_n Z_n$ mengikuti frame sebelumnya. Dan mengasumsikan *end-effector* adalah sendi *revolute*.

2.3.2 Transformasi Homogen

Transformasi Homogen adalah matriks untuk merepresentasikan nilai posisi dan rotasi dari sebuah *link*. Rotasi *link* memiliki tiga buah karakteristik yaitu rotasi terhadap sumbu x , y dan z . Matriks rotasi terhadap sumbu x , y dan z masing – masing dituliskan pada Persamaan 2.1, 2.2 dan 2.3.

$$Rot_{x,\theta} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos_{\theta} & -\sin_{\theta} & 0 \\ 0 & \sin_{\theta} & \cos_{\theta} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.1)$$

$$Rot_{y,\theta} = \begin{bmatrix} \cos_{\theta} & 0 & \sin_{\theta} & 0 \\ 0 & 1 & 0 & 0 \\ -\sin_{\theta} & 0 & \cos_{\theta} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$

$$Rot_{z,\theta} = \begin{bmatrix} \cos_{\theta} & -\sin_{\theta} & 0 & 0 \\ \sin_{\theta} & \cos_{\theta} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

Sedangkan untuk matriks translasi terhadap sumbu x , y dan z masing – masing dituliskan pada Persamaan 2.4, 2.5 dan 2.6.

$$Trans_{x,a} = \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

$$Trans_{y,b} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

$$Trans_{z,c} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

Matriks rotasi dan translasi tersebut akan membentuk matriks Transformasi Homogen yang terbentuk pada Persamaan 2.7. Notasi R berisi matriks rotasi sedangkan notasi d berisi matriks translasi.

$$H = \begin{bmatrix} R & d \\ 0 & 1 \end{bmatrix} \quad (2.7)$$

Sedangkan untuk pencarian *inverse* dari Transformasi Homogen ini diberikan oleh Persamaan 2.8.

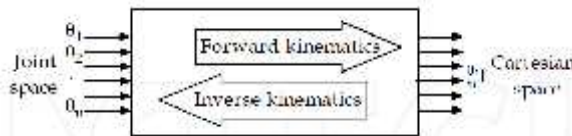
$$H^{-1} = \begin{bmatrix} R^T & -R^T d \\ 0 & 1 \end{bmatrix} \quad (2.8)$$

Transformasi Homogen yang paling populer dituliskan pada Persamaan 2.9. Masing – masing rotasi dan translasi digunakan notasi x , y dan z .

$$H = \begin{bmatrix} n & o & a & p \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9)$$

2.4 Kinematika

Kinematika adalah ilmu yang mempelajari pergerakan robot tanpa memperhitungkan gaya – gaya pada robot itu sendiri. Kinematika terdiri dari kinematika maju (*forward kinematics*) dan kinematika terbalik (*inverse kinematics*). Ada masukan sudut dan posisi yang diperhitungkan pada kedua kinematika tersebut. Gambar 2.6 menjelaskan bagaimana perbedaan antara kinematika maju dan kinematika terbalik.



Gambar 2.6 Perbedaan antara kinematika maju dan kinematika terbalik

2.4.1 Kinematika Maju (Forward Kinemaitcs)

Permasalahan dari kinematika maju terletak pada hubungan antara sendi pada robot manipulator dan posisi serta rotasi untuk *end-effector*. Secara formalnya, permasalahan dari kinematika maju adalah untuk menentukan posisi dan rotasi pada *end-effector* setelah diberikan nilai – nilai yang dibutuhkan untuk perumusan kinematika maju itu sendiri. Nilai – nilai tersebut didapat dari DH-parameters yang akan dibahas selanjutnya.

2.4.1.1 DH-Parameters

Dalam merepresentasikan nilai – nilai yang berada pada sebuah link dibutuhkan konvensi, yang paling terkenal sampai saat ini adalah *DH-convention* atau *Denavit-Hartenberg* [3]. Nilai – nilai yang terdapat pada *DH-convention* antara lain θ_i (*joint angle*), a_i (*link length*), d_i (*link offset*) dan α_i (*link twist*). Berdasarkan pembentukan sendi yang telah ditentukan sebelumnya pada sub bab 2.3.1 maka dapat ditentukan nilai tersebut dengan keterangan seperti berikut.

a_i = jarak sepanjang x_{i-1} dari z_{i-1} ke perpotongan dari sumbu x_i dan z_{i-1}

d_i = jarak sepanjang z_{i-1} dari u_{i-1} ke perpotongan dari sumbu x_i dan z_{i-1} . d_i dapat variabel jika sendi i adalah sendi *prismatic*.

α_i = sudut antara z_i dan z_{i-1} diukur terhadap x_i

θ_i = sudut antara x_i dan x_{i-1} diukur terhadap z_i . θ_i dapat variabel jika sendi i adalah *revolute*.

Setelah didapatkan nilai – nilai parameternya, maka dapat dibentuk matriks Transformasi Homogen dengan mensubstitusikan nilai – nilai tersebut. Matriks didasarkan terhadap rotasi dan translasi terhadap sumbu x dan z . Persamaan matriks Transformasi Homogen berdasarkan DH-Parameter suatu robot dapat dilihat pada Persamaan 2.10.

$$\begin{aligned}
 {}^{i-1}T_i &= Rot_{z,n_i}.Trans_z,d_i.Trans_x,a_i.Rot_x,\Gamma_i \\
 &= \begin{bmatrix} c_{n_i} & -s_{n_i} & 0 & 0 \\ s_{n_i} & c_{n_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\Gamma_i & -s\Gamma_i & 0 \\ 0 & s\Gamma_i & c\Gamma_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} c_{n_i} & -s_{n_i}c\Gamma_i & s_{n_i}s\Gamma_i & a_ic_{n_i} \\ s_{n_i} & c_{n_i}c\Gamma_i & -c_{n_i}s\Gamma_i & a_is_{n_i} \\ 0 & s\Gamma_i & c\Gamma_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R_i & p_i \\ 0 & 1 \end{bmatrix} \quad (2.10)
 \end{aligned}$$

2.4.1.2 Penghitungan kinematika maju

Konsep dari kinematika maju adalah menghitung nilai tujuan dari sudut yang telah diberikan seperti diilustrasikan melalui Persamaan 2.11.

$$n = (n_1, n_2, n_3, n_4, n_5, n_6, \dots, n_n) \quad (2.11)$$

Dari sudut – sudut yang telah diberikan tersebut akan didapatkan nilai akhir dari posisi robot. Nilai – nilai ini akan direpresentasikan sesuai matriks yang berisi seperti Persamaan 2.12, dimana notasi n , s dan a berisi nilai rotasi robot sedangkan notasi p berisi nilai translasi robot.

$$Y = (n, s, a, p) \quad (2.12)$$

Sudut yang telah diberikan akan menjadi nilai akhir untuk posisi robot. Nilai akhir tersebut dimasukkan masing – masing ke Persamaan 2.12 menggunakan perkalian dari matriks Transformasi Homogen untuk setiap link seperti yang tertera pada Persamaan 2.13.

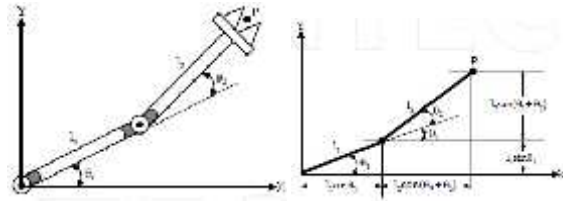
$${}^0_nT = {}^0_1T \times {}^1_2T \times {}^2_3T \times {}^3_4T \times {}^4_5T \times {}^5_6T \times \dots \times {}^{n-1}_nT \quad (2.13)$$

2.4.2 Kinematika Terbalik (*Inverse Kinematics*)

Pada umumnya, perhitungan untuk kinematika terbalik lebih sulit bila dibandingkan dengan kinematika maju. Karena permasalahan pada kinematika terbalik adalah mencari sudut dari setiap sendi berdasar pada posisi akhir yang diinginkan sebuah robot. Berdasarkan dari matriks yang diberikan untuk kemudian diproses menjadi sudut – sudut pembentuk setiap sendi robot. Sehingga kinematika terbalik dikatakan solusi yang mempunyai hasil akhir yang unik. Dikatakan unik karena hasil dari sudut untuk tiap sendi ada beberapa macam solusi tergantung dari banyaknya sendi yang digunakan. Metode untuk mencari kinematika terbalik sendiri ada berbagai macam.

2.4.2.1 Pendekatan Geometri

Pendekatan geometri didasarkan pengkomposisian ruang geometri dari robot manipulator ke beberapa permasalahan untuk cakupan geometrinya. Seperti yang terapkan ke robot manipulator 2-DOF seperti pada Gambar 2.8.



Gambar 2.7 Pengaplikasian pendekatan Geometri pada robot manipulator 2-DOF

Dari gambar tersebut dapat diambil Persamaan untuk mencari nilai p_x dan p_y . Yang masing – masing dijelaskan pada Persamaan 2.14 dan 2.15.

$$p_x = l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2) \quad (2.14)$$

$$p_y = l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2) \quad (2.15)$$

Persamaan diatas menunjukkan bagaimana mendapatkan perhitungan berdasar dari bentuk geometri robot. Berdasarkan dari Persamaan tersebut akan didapatkan θ_1 dan θ_2 .

2.4.2.2 Pendekatan Aljabar

Robot manipulator yang mempunyai banyak link dan sulit untuk dipecahkan menggunakan pendekatan geometri maka pendekatan aljabar lah solusi terbaiknya. Langkah – langkah dalam mencari solusi kinematika terbalik menggunakan pendekatan aljabar bagi robot manipulator yang mempunyai 6-DoF adalah sebagai berikut.

- a. Bentuk Persamaan *Direct Kinematics* yang terlihat pada Persamaan 2.16.

$$H = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^0T({}_1) {}^1T({}_2) {}^2T({}_3) {}^3T({}_4) {}^4T({}_5) {}^5T({}_6) \quad (2.16)$$

- b. Mencari solusi kinematika terbalik pada sendi pertama sebagai fungsi dari ${}_{end-effector}^{base}T$, maka transformasi kinematika terbalik dari link tersebut menjadi Persamaan 2.17.

$$\left[{}^0T({}_1) \right]^{-1} {}^0T = \left[{}^0T({}_1) \right]^{-1} {}^0T({}_1) {}^1T({}_2) {}^2T({}_3) {}^3T({}_4) {}^4T({}_5) {}^5T({}_6) \quad (2.17)$$

dimana $\left[{}^0T({}_1) \right]^{-1} {}^0T({}_1) = I$ sedangkan I adalah matriks identitas. Maka Persamaan 2.17 menjadi Persamaan 2.18.

$$\left[{}^0T({}_1) \right]^{-1} {}^0T = {}^1T({}_2) {}^2T({}_3) {}^3T({}_4) {}^4T({}_5) {}^5T({}_6) \quad (2.18)$$

- c. Berdasarkan algoritma yang dikembangkan [1], pencarian kinematika terbalik dirumuskan untuk mencari sendi ketiga saja yang akan didapatkan keseluruhan nilai masing – masing sendi. Maka persamaan yang dipakai berdasarkan Persamaan 2.19.

$${}^3T({}_4) {}^4T({}_5) {}^5T({}_6) = {}^2T^{-1}({}_3) {}^1T^{-1}({}_2) {}^0T^{-1}({}_1) {}^0T \quad (2.19)$$

- d. Dari Persamaan 2.19 tersebut akan didapat matriks untuk ruas kiri dan kanan yang terlihat pada Persamaan 2.20 dan 2.21.

$${}^3T_4 {}^4T_5 {}^5T_6 = \begin{bmatrix} R_4 R_5 R_6 & R_4 R_5 p_6 + R_4 p_5 + p_4 \\ 0 & 1 \end{bmatrix} \quad (2.20)$$

$${}^2T_3^{-1} {}^1T_2^{-1} {}^0T_1^{-1} {}^6T_0 = \begin{bmatrix} R_3^T R_2^T R_1^T R & R_3^T R_2^T R_1^T p - R_3^T R_2^T R_1^T p_1 - R_3^T R_2^T p_2 - R_3^T p_3 \\ 0 & 1 \end{bmatrix} \quad (2.21)$$

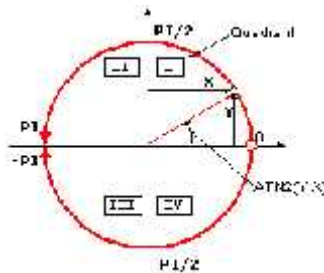
- e. Hasil terakhir akan didapatkan Persamaan 2.22 dan 2.23. yang berfungsi untuk pencarian sudut masing – masing sendi.

$$R_4 R_5 R_6 = R_3^T R_2^T R_1^T R \quad (2.22)$$

$$R_4 R_5 p_6 + R_4 p_5 + p_4 = R_3^T R_2^T R_1^T p - R_3^T R_2^T R_1^T p_1 - R_3^T R_2^T p_2 - R_3^T p_3 \quad (2.23)$$

2.4.2.3 Menghitung theta

Dari Persamaan 2.22 dan 2.23 akan didapatkan beberapa persamaan yang dapat disubstitusikan ataupun dieliminasi sehingga akan didapat persamaan sederhana untuk menghitung nilai sudut tertentu. Tabel 2.1 menunjukkan bagaimana persamaan yang sederhana akan didapatkan solusi mencari nilai sudut. Beberapa solusi memakai tanda \mp , yang mempunyai arti bahwa solusi tersebut akan menghasilkan dua hasil, tetapi berbeda dengan solusi yang mempunyai penghubung atau yang menandakan bahwa solusi hanya mempunyai satu solusi. Pada kasus pemecahan yang mempunyai satu solusi, dikarenakan bahwa karakteristik untuk nilai *tangent* sama pada kuadran 1 dan 3 yaitu notasi plus (+) seperti diilustrasikan pada Gambar 2.8, maka solusi yang didapat hanya satu.



Gambar 2.8 Fungsi *Atan2* pada empat kuadran

Tabel 2.1 Beberapa persamaan trigonometri yang digunakan kinematika terbalik

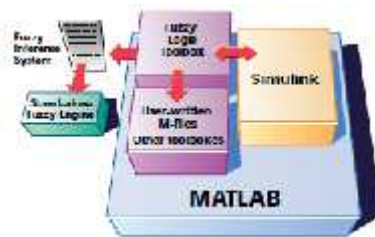
Persamaan	Solusi
$a \sin \theta + b \cos \theta = c$	$\theta = A \tan 2(a, b) \mp A \tan 2(\sqrt{a^2 + b^2 - c^2}, c)$
$a \sin \theta + b \cos \theta = 0$	$\theta = A \tan 2(-b, a)$ atau $\theta = A \tan 2(b, -a)$
$\cos \theta = a$ dan $\sin \theta = b$	$\theta = A \tan 2(b, a)$
$\sin \theta = b$	$\theta = A \tan 2(\mp \sqrt{1 - a^2}, a)$
$\sin \theta = a$	$\theta = A \tan 2(a, \mp \sqrt{1 - a^2})$

2.5 Sistem Fuzzy

Adalah salah satu bentuk pendekatan sistem manusia, dimana representasi suatu kejadian didistribusikan kedalam sejumlah istilah bahasa (yang menyatakan level kualitatif) dengan nilai kebenaran tiap level terletak antara 0 sampai 1. Sistem *fuzzy* diperkenalkan oleh Dr. Lotfi Zadeh dari Universitas California, Berkeley pada tahun 1965.

Motivasi utama teori sistem *fuzzy* adalah memetakan sebuah ruang masukan (*input*) ke dalam ruang keluaran (*output*) dengan menggunakan *IF-THEN rules*. Perkembangan dari sistem *fuzzy* sendiri memiliki banyak macam, antara lain *fuzzy* tipe Mamdani, Takagi-Sugeno dan lain - lain. Pada subbab ini akan dijelaskan mengenai himpunan *fuzzy*, fungsi keanggotaan *fuzzy*, operasi himpunan *fuzzy*, sistem inferensi *fuzzy* dan model *fuzzy* Mamdani.

Software MATLAB R2012a mempunyai *Toolbox* yang mendukung untuk menyusun sistem *fuzzy*. *Toolbox* ini berdiri sendiri (*stand alone*) yang mengizinkan *user* untuk mengakses secara langsung melalui bahasa C serta dapat juga diakses melalui *Simulink*. Gambaran umum *Toolbox* ini dalam ruang kerja MATLAB diilustrasikan seperti Gambar 2.9.



Gambar 2.9 Tata letak *Fuzzy Logic Toolbox* pada MATLAB

2.5.1 Himpunan *Fuzzy*

Adalah sebuah himpunan dimana keanggotaan dari tiap elemennya tidak mempunyai batas yang jelas atau suatu himpunan yang beranggotakan sejumlah istilah dalam pengertian bahasa yang menyatakan level kualitatif dari semesta pembicaraan X . Himpunan *fuzzy* juga merupakan cara untuk menerjemahkan tentang beberapa masukan ke dalam suatu bentuk fungsi keanggotaan. Masing – masing elemen mempunyai derajat keanggotaan untuk menjadi anggota himpunan, dimana fungsi keanggotaan dari suatu himpunan *fuzzy* bernilai $0,0 - 1,0$.

2.5.2 Fungsi Keanggotaan *Fuzzy*

Fungsi keanggotaan mendefinisikan bagaimana tiap titik dalam ruang masukan (*input*) dipetakan menjadi bobot atau derajat keanggotaan antara 0 dan 1. Dalam teori himpunan, ruang masukan juga dikenal sebagai *universe of discourse*. Misalkan ada dua *fuzzy set*, yaitu A dan B serta ada sebuah anggota himpunan X . Anggota himpunan X ini bisa dikelompokkan dalam *fuzzy set* B dengan derajat keanggotaan $\sim_a(x)$ dan juga ada *fuzzy set* B dengan derajat keanggotaan $\sim_b(x)$. Sekarang lihat jika operasi min dilakukan antara dua *fuzzy set* A dan B tersebut, yang dinotasikan pada Persamaan 2.24.

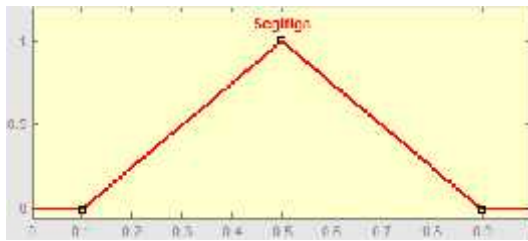
$$\min(\sim_a(x), \sim_b(x)) \quad (2.24)$$

2.5.2.1 Fungsi Keanggotaan Segitiga

Fungsi keanggotaan segitiga ditunjukkan pada Gambar 2.9. Secara matematis, derajat keanggotaan untuk fungsi keanggotaan

segitiga ditentukan dengan Persamaan 2.25 sedangkan ilustrasinya terlihat pada Gambar 2.10.

$$\sim(x) = \begin{cases} 0, & x < a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & x > c \end{cases} \quad (2.25)$$

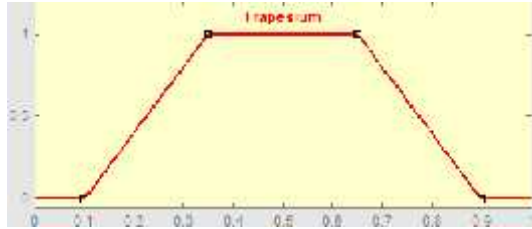


Gambar 2.10 Fungsi Keanggotaan Segitiga

2.5.2.2 Fungsi Keanggotaan Trapesium

Persamaan fungsi keanggotaan trapesium (*trapezoid*) dengan empat parameter skalar $\{a, b, c, d\}$ dengan $a < b < c < d$. Fungsi keanggotaan trapesium ditunjukkan pada Gambar 2.11. Derajat keanggotaannya dituliskan pada Persamaan 2.26

$$\sim(x) = \begin{cases} 0, & x < a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \\ 0, & x > d \end{cases} \quad (2.26)$$



Gambar 2.11 Fungsi Keanggotaan Trapesium

2.5.2.3 Fungsi Keanggotaan Gaussian

Dua parameter $\{c, \sigma\}$ menentukan fungsi keanggotaan *Gaussian* dan derajat keanggotaannya ditentukan pada Persamaan 2.27.

$$\mu(x) = e^{-\frac{1}{2\sigma^2}(x-c)^2} \quad (2.27)$$

dengan c merepresentasikan pusat fungsi keanggotaan dan σ (standar deviasi) mendefinisikan lebar fungsi keanggotaan. Berikut fungsi keanggotaan *Gaussian* yang ditunjukkan pada Gambar 2.12. Nilai dari *Gaussian* tidak akan mencapai 0, karena dari Persamaan 2.27 terlihat bahwa persamaan tersebut nilainya tidak akan mencapai 0 apabila nilai c dan σ diisi dengan nilai berapapun kecuali nilai 0.



Gambar 2.12 Fungsi Keanggotaan *Gaussian*

2.5.3 Operasi Himpunan Fuzzy

Operasi himpunan *fuzzy* dilakukan dengan mengoperasikan fungsi keanggotaan satu dengan lainnya. Operasi ini dilakukan dalam proses interferensi *fuzzy*. Macam – macam operator *fuzzy* antara lain adalah operator *min* (minimum), *max* (maximum), dan *product*. Pada Gambar 2.13 terdapat himpunan *fuzzy* A dan B yang akan dilakukan operasi himpunan *fuzzy*. Dua himpunan *fuzzy* dalam semesta pembicaraan X dengan fungsi keanggotaan μ_A dan μ_B untuk $x \in X$, maka operasi himpunan *fuzzy* didefinisikan sebagai berikut:

1. Union (*OR*)

Union (gabungan) dari himpunan *fuzzy* A dan B dengan fungsi keanggotaan $\sim_A(x)$ dan $\sim_B(x)$ adalah $\sim_{A \cup B}(x)$.

2. Interseksi (*AND*)

Interseksi himpunan *fuzzy* A dan B adalah himpunan *fuzzy* yang fungsi keanggotaannya diberikan sebagai operasi *minimum* Persamaan 2.28 dan operasi *Algebraic product* Persamaan 2.29 dinyatakan sebagai berikut:

$$\sim_{A \cap B}(x) = \min\{\sim_A(x), \sim_B(x)\}, x \in X \quad (2.28)$$

$$\sim_{A \cap B}(x) = \sim_A(x) \cdot \sim_B(x), x \in X \quad (2.29)$$

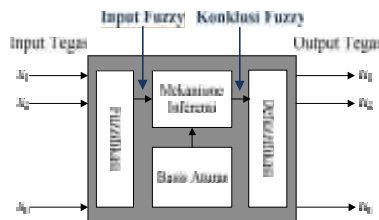
3. Komplemen (*NOT*)

Komplemen dari suatu himpunan *fuzzy* B dengan fungsi keanggotaan $\sim_B(x)$ pada Persamaan (2.30) berikut:

$$\sim_{B^c}(x) = 1 - \sim_B(x) \text{ untuk } x \in X \quad (2.30)$$

2.5.4 Sistem Inferensi Fuzzy

Proses penarikan kesimpulan dari sekumpulan kaidah *fuzzy* atau masukan tegas. Dalam sistem *fuzzy* yang diterangkan pada Gambar 2.13, terlihat bahwa sistem inferensi *fuzzy* dalam membuat sebuah keputusan terlihat amat rumit dengan melalui beberapa proses terlebih dahulu.



Gambar 2.13 Sistem Inferensi Fuzzy

Sistem sistem Inferensi *fuzzy* mempunyai kemiripan dengan penalaran manusia seperti pengetahuan yang meliputi penalaran *fuzzy*. Penalaran *fuzzy* tersebut dinyatakan sebagai aturan dalam bentuk pernyataan **If-Then** dinyatakan dengan

$$\underbrace{\text{If } x_1 \text{ is } A \text{ and } x_2 \text{ is } B}_{\text{premis}} \text{ Then } \underbrace{u = C}_{\text{konsekuen}}$$

Dalam menginterpretasikan sebuah **If-Then rule** meliputi dua bagian. Pertama, mengevaluasi *antecedent*, melakukan fuzzifikasi pada masukan dan menerapkan operasi – operasi sistem *fuzzy* dengan operator – operator *fuzzy*. Kedua, menerapkan hasil operasi sistem *fuzzy* pada bagian *antecedent* untuk mengambil kesimpulan dengan mengisikan *fuzzy set* keluaran ke variabel keluaran.

Proses inferensi untuk menentukan keputusan dalam suatu sistem sistem *fuzzy* adalah sebagai berikut:

1. Fuzzifikasi input
Fuzzifikasi masing-masing input tegas pada fungsi keanggotaan himpunan *fuzzy* sehingga didapat derajat keanggotaanya.
2. Aplikasi operator *fuzzy*
Penerapan operator *fuzzy* (seperti *AND*, *OR*) pada masing-masing bagian premis dari aturan *fuzzy*, dimana operasi *AND* berarti mengambil nilai minimum sedangkan *OR* nilai maksimum.
3. Aplikasi implikasi
Proses mendapatkan konsekuen dari aturan (**If-Then**) berdasarkan derajat keanggotaan premis.
4. Agregasi
Proses penggabungan semua hasil implikasi *fuzzy* dari setiap aturan (**If-Then**) menjadi sebuah *fuzzy set* tunggal.
5. Defuzzifikasi
Perhitungan keputusan *fuzzy* sehingga didapat output tegas.

Toolbox pada MATLAB yang digunakan telah mencakup keseluruhan dari sistem inferensi *fuzzy*. *Toolbox* ini telah dapat menangani berapa pun masukan dan keluaran dari sistem *fuzzy* yang diharapkan. Hanya saja ketergantungan berapa masukan dan keluaran yang diijinkan tergantung dari performansi komputer yang digunakan. Gambar 2.14 menunjukkan tampilan keseluruhan.



Gambar 2.14 FIS Editor pada Toolbox Fuzzy MATLAB

Metode defuzzifikasi yang sering digunakan diantaranya adalah *center of gravity*, dan *center (weighted) average*. Metode defuzzifikasi dengan *center of gravity* dapat diterapkan pada fungsi keanggotaan yang simetris. Sistem inferensi *fuzzy* dapat dibangun dengan dua metode, yaitu metode Mamdani dan metode Sugeno. Dalam mekanisme inferensi, perbedaan sistem inferensi *fuzzy* tipe Mamdani dan Sugeno terletak pada jenis fungsi keanggotaan yang digunakan pada bagian konsekuen.

2.5.5 Metode Sugeno

Metode ini banyak mempunyai kelebihan apabila dibandingkan dengan tipe Mamdani [8]. Salah satunya dapat menghitung luas daerah di bawah kurva. Hanya saja ada kemiripan pada proses fuzzifikasi, operasi *fuzzy* dan implikasi. Perbedaan terletak pada proses agregasi dan defuzzifikasi, jika pada Mamdani agregasi berupa daerah di bawah kurva maka pada metode sugeno agregasi berupa *singleton – singleton*. Defuzzifikasi pada metode sugeno menggunakan *center of weighed area* yang terlihat pada Persamaan 2.31.

$$z^* = \frac{\sum \tilde{c}_c(z) \cdot z}{\sum \tilde{c}_c(z)} \quad (2.31)$$

Dalam hal ini, z adalah nilai *singleton*.

BAB III DESAIN SISTEM

Pada Bab ini dijelaskan mengenai pemodelan robot Denso 6-DoF lengkap dengan langkah – langkah memasukkan nilai dari DH-Parameter ke Transformasi Homogen, perancangan kinematika terbalik (*inverse kinematics*) sampai desain sistem *fuzzy*. Perhitungan kinematika terbalik yang dipakai menggunakan metode aljabar sedangkan sistem *fuzzy* yang dipakai adalah tipe Sugeno.

3.1 Pemodelan Transformasi Homogen Robot Denso 6-DoF

Menurut penelitian tentang DH-Parameter [3], pemodelan keseluruhan rancangan untuk sebuah robot dapat didefinisikan menjadi θ_t (*joint angle*), a_t (*link length*), d_t (*link offset*) dan α_t (*link twist*). Nilai – nilai yang didapat dari penentuan keempat parameter tersebut akan dimasukkan ke model matriks Transformasi Homogen.

3.1.1 DH-Parameter Robot Denso 6-DoF

Pemodelan DH-Parameter untuk robot Denso 6-DoF dapat dijabarkan sesuai dengan Tabel 3.1 [6].

Tabel 3.1 DH-Parameter robot Denso 6-DOF

<i>Link</i>	a_t	d_t	α_t	θ_t
1	75	335	90	$\pm 170^\circ$
2	270	0	0	$+ 135^\circ, - 100^\circ$
3	90	0	-90	$+ 166^\circ, - 119^\circ$
4	0	295	90	$\pm 190^\circ$
5	0	0	-90	$\pm 120^\circ$
6	0	108	0	$\pm 360^\circ$

Dari DH-Parameter inilah yang akan digunakan untuk memodelkan Matriks Transformasi Homogen untuk perumusan perhitungan kinematika maju dan terbalik.

3.1.2 Model Transformasi Homogen tiap *Link*

Berdasarkan [5], dari pemodelan DH-Parameter dapat dimasukkan nilai tersebut seperti pada persamaan 2.10 masing – masing seperti berikut ini

$${}^0_1T = \begin{bmatrix} c_{n_1} & 0 & s_{n_1} & a_1 c_{n_1} \\ s_{n_1} & 0 & -c_{n_1} & a_1 s_{n_1} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.1)$$

$${}^1_2T = \begin{bmatrix} c_{n_2} & -s_{n_2} & 0 & a_2 c_{n_2} \\ s_{n_2} & c_{n_2} & 0 & a_2 s_{n_2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

$${}^2_3T = \begin{bmatrix} c_{n_3} & 0 & -s_{n_3} & a_3 c_{n_3} \\ s_{n_3} & 0 & c_{n_3} & a_3 s_{n_3} \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.3)$$

$${}^3_4T = \begin{bmatrix} c_{n_4} & 0 & -s_{n_4} & 0 \\ s_{n_4} & 0 & c_{n_4} & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.4)$$

$${}^4_5T = \begin{bmatrix} c_{n_5} & 0 & -s_{n_5} & 0 \\ s_{n_5} & 0 & c_{n_5} & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.5)$$

$${}^5_6T = \begin{bmatrix} c_{n_6} & -s_{n_6} & 0 & 0 \\ s_{n_6} & c_{n_6} & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.6)$$

3.2 Pemodelan Robot Denso 6-DoF pada SOLIDWORK

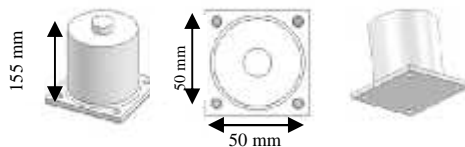
Perancangan pada *Software* SOLIDWORK bertujuan agar saat jalannya program dapat dianalisa gerakan sebenarnya pada robot Denso 6-DoF. Hal – hal yang harus diperhatikan dalam memodelkan robot ini adalah panjang dari setiap *link* yang dimiliki robot. Dari *manual* robot Denso 6-DoF [9], didapat perancangan panjang setiap *link* sesuai pada Gambar 3.1.



Gambar 3.1 Panjang setiap *link* dalam satuan milimeter

3.2.1 Rancangan *Base Frame* dan setiap *Link*

Pemodelan yang dilakukan pada Solidwork nantinya akan dijadikan tampilan saat jalannya program pada MATLAB. Dimulai dari *Base Frame* sampai pemodelan untuk end-effector dari robot Denso 6-DoF ini. *Base Frame* atau *Link* 1 dirancang secara simetris untuk bagian bawah dari robot sepanjang 50 mm. Tinggi d_1 sebesar 155 mm. Lebih lengkapnya dapat dilihat pada Gambar 3.2.



Gambar 3.2 Model *Base Frame* atau *Link* 1

Selanjutnya adalah pembuatan design untuk *Link* 2, dimana panjangnya sebesar 125 mm dari sendi pertama ke sendi kedua. Terlihat pada Gambar 3.3 untuk pemodelan *Link* 2.



Gambar 3.3 Model *Link 2*

Link 3 pada bagian *elbow* robot Denso 6-DoF berdasarkan pada DH-Parameter yang didapat untuk a_3 panjangnya sebesar 210 mm. Maka dapat digambarkan seperti pada Gambar 3.4.



Gambar 3.4 Model *Link 3*

Link 4 sampai 6 yang tidak termasuk pada bagian *elbow* melainkan *wrist* robot Denso 6-DoF tidak terdapat panjang a_i (*link length*). Sehingga untuk *Link 4* yang masih menjadi bagian *elbow* robot Denso 6-DoF, mempunyai d_4 panjangnya sebesar 88 mm. Seperti terlihat pada Gambar 3.5.



Gambar 3.5 Model *Link 4*

Link 5 yang tidak mempunyai ukuran untuk a_5 dan d_5 lebih jelasnya terlihat pada Gambar 3.6.



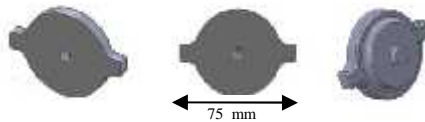
Gambar 3.6 Model *Link 5*

Sedangkan untuk *Link* terakhir atau *Link 6*, mempunyai d_6 sepanjang 108 mm. Pemodelan untuk *Link 6* sendiri dapat dilihat pada Gambar 3.7.



Gambar 3.7 Model *Link 6*

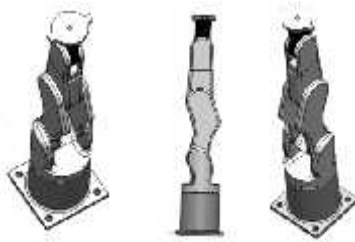
Perancangan *end-effector* dibentuk dengan model bundar dengan diameter bundaran yang dibentuk sebesar 100 mm. Pemodelan *end-effector* dapat dilihat pada Gambar 3.8.



Gambar 3.8 Model *end-effector*

3.2.2 Rancangan keseluruhan Robot

Dari keseluruhan bagian robot yang dimodelkan pada SOLIDWORK, selanjutnya keseluruhan bagian ini disatukan (*assembly*) untuk pemodelan keseluruhan robot Denso 6-DoF. Pemodelan keseluruhan ini ditunjukkan pada Gambar 3.9.



Gambar 3.9 Pemodelan robot Denso 6-DoF

3.3 Kinematika

Perhitungan untuk mendapatkan persamaan kinematika maju dan kinematika terbalik menggunakan software MATLAB untuk pemrosesan matriks dan mendapatkan solusi beberapa *theta* dari persamaan matriks itu sendiri. Sebelum memasuki perhitungan tersebut, penulis menyederhanakan beberapa simbol, yaitu

$$s_i = \sin(\theta_i) \quad (3.7)$$

$$c_i = \cos(\theta_i) \quad (3.8)$$

$$s_{ij} = \sin(\theta_i + \theta_j) \quad (3.9)$$

$$c_{ij} = \cos(\theta_i + \theta_j) \quad (3.10)$$

$$\sin(\theta_i + \theta_j) = (\sin \theta_i \cos \theta_j + \cos \theta_i \sin \theta_j) \quad (3.11)$$

$$\cos(\theta_i + \theta_j) = (\cos \theta_i \cos \theta_j - \sin \theta_i \sin \theta_j) \quad (3.12)$$

3.3.1 Kinematika Maju (*Forward Kinematics*)

Pemodelan kinematika maju yang mempunyai 6 matriks dapat dihitung berdasarkan Persamaan 3.13. Hasil akhir dari perhitungan akan didapatkan matriks:

$$\begin{bmatrix} n & o & a & p \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.13)$$

Setelah perhitungan perkalian antara beberapa *Link* model Persamaan 3.1 sampai 3.6, maka didapatkan hasil Persamaan 3.14 sampai 3.25.

$$n_x = -c_{n6}(s_{n5}(c_{n1}c_{n2}s_{n3} + c_{n1}c_{n3}s_{n2}) - c_{n5}(c_{n4}(c_{n1}c_{n2}c_{n3} - c_{n1}s_{n2}s_{n3}) - s_{n1}s_{n4})) - s_{n6}(s_{n4}(c_{n1}c_{n2}c_{n3} - c_{n1}s_{n2}s_{n3}) + c_{n4}s_{n1})) \quad (3.14)$$

$$o_x = s_{n6}(s_{n5}(c_{n1}c_{n2}s_{n3} + c_{n1}c_{n3}s_{n2}) - c_{n5}(c_{n4}(c_{n1}c_{n2}c_{n3} - c_{n1}s_{n2}s_{n3}) - s_{n1}s_{n4})) - c_{n6}(s_{n4}(c_{n1}c_{n2}c_{n3} - c_{n1}s_{n2}s_{n3}) + c_{n4}s_{n1})) \quad (3.15)$$

$$a_x = -c_{n5}(c_{n1}c_{n2}s_{n3} + c_{n1}c_{n3}s_{n2}) - s_{n5}(c_{n4}(c_{n1}c_{n2}c_{n3} - c_{n1}s_{n2}s_{n3}) - s_{n1}s_{n4})) \quad (3.16)$$

$$p_x = a_1c_{n1} - d_6(c_{n5}(c_{n1}c_{n2}s_{n3} + c_{n1}c_{n3}s_{n2}) + s_{n5}(c_{n4}(c_{n1}c_{n2}c_{n3} - c_{n1}s_{n2}s_{n3}) - s_{n1}s_{n4})) - d_4(c_{n1}c_{n2}s_{n3} + c_{n1}c_{n3}s_{n2}) + a_2c_{n1}c_{n2} + a_3c_{n1}c_{n2}c_{n3} - a_3c_{n1}s_{n2}s_{n3} \quad (3.17)$$

$$a_y = -c_{n5}(c_{n2}s_{n1}s_{n3} + c_{n3}s_{n1}s_{n2}) - s_{n5}(c_{n4}(c_{n2}c_{n3}s_{n1} - s_{n1}s_{n2}s_{n3}) + c_{n1}s_{n4})) \quad (3.18)$$

$$p_y = a_1s_{n1} - d_4(c_{n2}s_{n1}s_{n3} + c_{n3}s_{n1}s_{n2}) - d_6(c_{n5}(c_{n2}s_{n1}s_{n3} + c_{n3}s_{n1}s_{n2}) + s_{n5}(c_{n4}(c_{n2}c_{n3}s_{n1} - s_{n1}s_{n2}s_{n3}) + c_{n1}s_{n4})) + a_2c_{n2}s_{n1} + a_3c_{n2}c_{n3}s_{n1} - a_3s_{n1}s_{n2}s_{n3} \quad (3.19)$$

$$n_y = -c_{n6}(s_{n5}(c_{n2}s_{n1}s_{n3} + c_{n3}s_{n1}s_{n2}) - c_{n5}(c_{n4}(c_{n2}c_{n3}s_{n1} - s_{n1}s_{n2}s_{n3}) + c_{n1}s_{n4})) - s_{n6}(s_{n4}(c_{n2}c_{n3}s_{n1} - s_{n1}s_{n2}s_{n3}) - c_{n1}c_{n4})) \quad (3.20)$$

$$o_y = s_{n6}(s_{n5}(c_{n2}s_{n1}s_{n3} + c_{n3}s_{n1}s_{n2}) - c_{n5}(c_{n4}(c_{n2}c_{n3}s_{n1} - s_{n1}s_{n2}s_{n3}) + c_{n1}s_{n4})) - c_{n6}(s_{n4}(c_{n2}c_{n3}s_{n1} - s_{n1}s_{n2}s_{n3}) - c_{n1}c_{n4})) \quad (3.21)$$

$$n_z = s_{n4}s_{n6}(c_{n2}s_{n3} + c_{n3}s_{n2}) - c_{n6}(s_{n5}(c_{n2}c_{n3} - s_{n2}s_{n3}) + c_{n4}c_{n5}(c_{n2}s_{n3} + c_{n3}s_{n2})) \quad (3.22)$$

$$o_z = s_{n6}(s_{n5}(c_{n2}c_{n3} - s_{n2}s_{n3}) + c_{n4}c_{n5}(c_{n2}s_{n3} + c_{n3}s_{n2})) + c_{n6}s_{n4}(c_{n2}s_{n3} + c_{n3}s_{n2}) \quad (3.23)$$

$$a_z = c_{n4}s_{n5}(c_{n2}s_{n3} + c_{n3}s_{n2}) - c_{n5}(c_{n2}c_{n3} - s_{n2}s_{n3}) \quad (3.24)$$

$$p_z = d_1 - a_2 s_{n2} - d_4 (c_{n2} c_{n3} - s_{n2} s_{n3}) - d_6 (c_{n5} (c_{n2} c_{n3} - s_{n2} s_{n3}) - c_{n4} s_{n5} (c_{n2} s_{n3} + c_{n3} s_{n2})) - a_3 c_{n2} s_{n3} - a_3 c_{n3} s_{n2} \quad (3.25)$$

Karena perhitungan dari kinematika maju menggunakan masukan dari masing – masing *theta* yang telah diberikan, maka *theta* harus didefinisikan terlebih dahulu untuk mendapatkan perubahan orientasi robot.

3.3.2 Kinematika Terbalik (*Inverse Kinematics*)

Masukan yang berbalik dari kinematika maju ini mengambil konsep perhitungan aljabar yang telah dibahas [1]. Dengan menggunakan Persamaan 2.22 maka akan didapatkan matriks persamaan 3.26 untuk ruas kiri dan 3.27 untuk ruas kanan:

$$\begin{bmatrix} s_{23}n_z + c_{23}(n_x c_1 + n_y s_1) & s_{23}o_z + c_{23}(o_x c_1 + o_y s_1) & s_{23}a_z + c_{23}(a_x c_1 + a_y s_1) \\ n_y c_1 - n_x s_1 & o_y c_1 - o_x s_1 & a_y c_1 - a_x s_1 \\ c_{23}n_z + s_{23}(-n_x c_1 - n_y s_1) & c_{23}o_z + s_{23}(-o_x c_1 - o_y s_1) & c_{23}a_z + s_{23}(-a_x c_1 - a_y s_1) \end{bmatrix} \quad (3.26)$$

$$\begin{bmatrix} c_4 c_5 c_6 - s_4 s_6 & -c_6 s_4 - c_4 c_5 s_6 & -c_4 s_5 \\ c_4 s_6 + c_5 c_6 s_4 & c_4 c_6 - c_5 s_4 s_6 & -s_4 s_5 \\ c_6 s_5 & -s_5 s_6 & c_5 \end{bmatrix} \quad (3.27)$$

Dari Persamaan 2.23 juga akan didapatkan persamaan Persamaan 3.28:

$$\begin{bmatrix} -d_6 c_4 s_5 \\ -d_6 s_4 s_5 \\ d_4 + d_6 c_5 \end{bmatrix} = \begin{bmatrix} s_{23}(p_z - d_1 - a_2 s_2) + c_{23}(-a_2 c_2 + p_x c_1 + p_y s_1 - a_1) - a_3 \\ p_y c_1 - p_x s_1 \\ c_{23}(p_z - d_1 - a_2 s_2) + s_{23}(a_2 c_2 - p_x c_1 - p_y s_1 + a_1) \end{bmatrix} \quad (3.28)$$

Dari kedua persamaan matriks tersebut hanya beberapa persamaan saja yang diambil. Diantaranya adalah sebagai berikut:

$$a_3 - d_6 c_4 s_5 = s_{23}(p_z - d_1 - a_2 s_2) + c_{23}(-a_2 c_2 + p_x c_1 + p_y s_1 - a_1) \quad (3.29)$$

$$-d_6 s_4 s_5 = p_y c_1 - p_x s_1 \quad (3.30)$$

$$d_4 + d_6 c_5 = c_{23}(p_z - d_1 - a_2 s_2) + s_{23}(a_2 c_2 - p_x c_1 - p_y s_1 + a_1) \quad (3.31)$$

$$c_{23}n_z + s_{23}(-n_x c_1 - n_y s_1) = c_6 s_5 \quad (3.32)$$

$$c_{23}o_z + s_{23}(-o_x c_1 - o_y s_1) = -s_5 s_6 \quad (3.33)$$

$$c_{23}a_z + s_{23}(-a_x c_1 - a_y s_1) = c_5 \quad (3.34)$$

$$a_y c_1 - a_x s_1 = -s_4 s_5 \quad (3.35)$$

$$s_{23}a_z + c_{23}(a_x c_1 + a_y s_1) = -c_4 s_5 \quad (3.36)$$

Dengan menggunakan Persamaan 3.29 sampai 3.36 dan cara penyelesaian mencari *theta* berdasarkan Tabel 2.1 akan didapatkan persamaan mencari masing – masing *theta* seperti dijabarkan berikut ini:

• **Theta 1**

$$\left(\frac{p_y}{d_6} - a_y \right) c_1 + \left(\frac{-p_x}{d_6} + a_x \right) s_1 = 0 \quad (3.37)$$

• **Theta 2**

Dengan asumsi bahwa

$$x_2 = d_1 - p_z$$

$$y_2 = c_1 p_x + s_1 p_y - a_1$$

$$z_2 = -\frac{(d_1 - p_z)^2}{2a_2} - \frac{(c_1 p_x + s_1 p_y - a_1)^2}{2a_2} - \frac{(a_2)^2}{2a_2} + \frac{(a_3)^2}{2a_2} + \frac{(d_4)^2}{2a_2} + \frac{d_6 (a_z)^2}{2a_2} + \frac{d_6 (a_x c_1 + a_y s_1)^2}{2a_2}$$

Maka didapat persamaan untuk *theta* 2:

$$(x_2)s_2 - (y_2)c_2 = z_2 \quad (3.38)$$

• **Theta 3**

Dengan asumsi bahwa

$$x_3 = \left(\frac{(p_z - d_1 - a_2 s_2)}{d_6} - a_z \right)$$

$$y_3 = \left(\frac{(-a_2 c_2 + p_x c_1 + p_y s_1 - a_1)}{d_6} - a_x c_1 - a_y s_1 \right)$$

$$z_3 = \frac{a_3}{d_6}$$

Maka didapat persamaan untuk menghitung *theta* 3:

$$\sin_{\theta_3} (\cos_{\theta_2} (x_3) - \sin_{\theta_2} (y_3)) + \cos_{\theta_3} (\sin_{\theta_2} (x_3) - \cos_{\theta_2} (y_3)) = z_3 \quad (3.39)$$

• **Theta 5**

$$c_{23}a_z + s_{23}(-a_x c_1 - a_y s_1) = c_5 \quad (3.40)$$

• **Theta 4**

$$a_y c_1 - a_x s_1 = -s_4 s_5 \quad (3.41)$$

• **Theta 6**

$$c_{23}n_z + s_{23}(-n_x c_1 - n_y s_1) = c_6 c_5 \quad (3.42)$$

3.4 Desain Sistem Fuzzy

Pada Tugas Akhir ini, digunakannya sistem *fuzzy* dimaksudkan untuk mendapatkan besarnya *theta* 1. Dengan memberi nilai masukan pada persamaan untuk mencari *theta* 1 sebelumnya, maka akan didapat beberapa hasil yang dijadikan referensi pembuatan desain sistem *fuzzy*. Beberapa hasil ini dapat dilihat pada Tabel 3.2.

Hasil yang didapat pada perhitungan yang ada pada Tabel 3.2 didapat dari persamaan untuk mencari *theta* 1 yang ada pada Persamaan 3.37. Nilai masukan yang digunakan mempunyai jarak dari -200 sampai 200, dimana jarak antar satu kemungkinan dengan kemungkinan yang lain bernilai 25. Digunakannya nilai 25 untuk jarak antar kemungkinan tersebut bertujuan agar semakin akuratnya hasil yang didapat.

Penggunaan sistem *fuzzy* karena secara konsepnya, sistem *fuzzy* mudah untuk dimengerti dan diimplementasikan serta akan menguntungkan dari segi fleksibilitas robot [2]. Dalam tahap desain kinematika terbalik untuk Robot Manipulator Denso 6-DoF dijelaskan bahwa untuk mendapatkan nilai keseluruhan *theta* setiap sendi langkah pertama yang dilakukan adalah menghitung *theta* 1. Berdasarkan tahap desain tersebut maka dapat dijelaskan bahwa nilai *theta* 1 berpengaruh terhadap nilai *theta* yang lain. Oleh karena itu sistem *fuzzy* yang akan dirancang dikhususkan untuk mencari nilai *theta* 1. Perancangan sistem *fuzzy* untuk *theta* 1 juga ditujukan untuk membandingkan keakuratan hasil yang dikeluarkan bila dibandingkan menggunakan perhitungan manual.

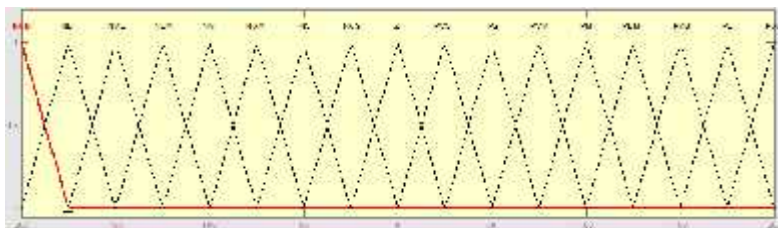
Tabel 3.2. Beberapa nilai Kemungkinan theta /

Theta I	$\rho \times$																
	-200	-175	-150	-125	-100	-75	-50	-25	0	25	50	75	100	125	150	175	200
-200	28.4448	36.5948	33.0771	35.8938	39.1796	43.0029	47.4632	52.6659	58.6997	65.6668	73.5787	81.9188	-89.3708	-86.5899	-72.2104	-64.5816	-57.79
-175	24.4021	26.5548	28.8157	31.4549	34.5618	38.2483	42.6126	47.9566	54.2714	61.7976	70.4833	80.3489	-89.2557	-78.9666	-69.2357	-60.6319	-53.5024
-150	20.5084	22.2691	24.1999	26.5535	29.3687	32.7802	36.9444	42.1534	48.6348	56.7179	66.0293	78.2886	-89.0889	-74.5055	-65.1028	-55.4029	-47.0244
-125	16.1839	17.5777	19.2237	21.0239	23.1677	26.5489	30.2815	35.0636	41.3849	49.7676	60.8738	74.9787	-88.8258	-72.8111	-59.1128	-48.4261	-40.7889
-100	11.6617	12.6956	13.9266	15.4156	17.2503	19.5615	22.551	26.5424	32.0727	40.0496	51.0213	65.3265	-88.1491	-66.4918	-49.9313	-38.7202	-31.1051
-75	6.9876	7.6198	8.3768	9.2993	10.4477	11.9151	13.8526	16.5217	20.4113	26.5269	37.1596	57.5672	-87.2214	-57.776	-35.2218	-25.4591	-19.7557
-50	2.2182	2.421	2.6645	2.9624	3.3532	3.8153	4.4563	5.3556	6.7075	8.9649	13.4707	26.4445	-81.7695	-23.3384	-12.578	-8.5569	-6.4757
-25	-2.3819	-2.8178	-3.101	-3.4475	-3.8811	-4.4391	-5.184	-6.2382	-7.7978	-10.4664	-15.3813	-30.0711	82.4872	26.6694	14.5695	9.9553	7.5372
0	-7.5481	-8.0099	-8.4048	-9.7726	-10.9768	-12.5141	-14.5415	-17.2386	-21.7365	-27.7021	-38.3638	-58.8843	87.3582	55.1478	36.598	26.6015	20.6955
25	-12.0108	-11.9734	-14.3381	-15.6869	-17.7488	-20.1164	-23.1735	-27.4835	-33.8599	-40.9079	-52.762	-69.8942	88.1984	67.1211	50.7853	39.5712	31.9493
50	-16.5193	-17.9377	-19.611	-21.6148	-24.0463	-27.0485	-30.8249	-35.6697	-42.0016	-50.3787	-61.5987	-75.2866	88.1509	73.1585	59.6169	49.0416	40.9927
75	-20.8273	-22.449	-24.562	-26.9406	-29.7832	-33.2215	-37.4294	-42.6342	-49.1132	-57.1584	-66.9499	-78.4298	89.1041	76.721	65.4866	55.9125	48.1048
100	-24.9025	-26.8721	-29.1566	-31.8078	-34.9318	-38.633	-43.0465	-48.3292	-54.645	-62.1254	-70.795	-80.7548	89.4658	79.0549	69.0462	60.9802	53.6802
125	-28.7257	-30.8885	-33.3615	-36.2119	-39.5074	-43.3464	-47.7861	-52.9088	-58.9957	-65.8873	-73.5811	-81.9124	89.7302	89.6971	72.4238	64.8398	58.0908
150	-32.3886	-34.5946	-37.205	-40.1711	-43.5514	-47.41	-51.8128	-56.8194	-62.4698	-68.7852	-75.6454	-82.9718	89.4696	81.9137	74.6376	67.8339	61.6281
175	-35.9922	-37.9075	-40.692	-43.7172	-47.1178	-50.9366	-55.2235	-60.0017	-65.3887	-71.0648	-77.5685	-83.7889	89.2558	82.8304	76.3656	70.2171	64.5078
200	-38.6442	-41.1116	-43.8489	-46.884	-50.2626	-54.0026	-58.1338	-62.6708	-67.6113	-72.9286	-78.5662	-84.5387	89.7377	83.5934	77.4466	72.1528	66.8863

Dengan mengacu pada nilai – nilai kemungkinan yang didapat, maka dapat didesain kontroller *fuzzy* yang dapat memproses beberapa kemungkinan tersebut. Sistem *fuzzy* ini sendiri mempunyai dua masukan dan satu keluaran. Masukan yang berupa nilai koordinat yang dituju untuk nilai pX dan pY . Langkah – langkah yang dilakukan untuk mendesain sistem *fuzzy* dalam mencari θ 1 sebagai berikut.

• Langkah 1

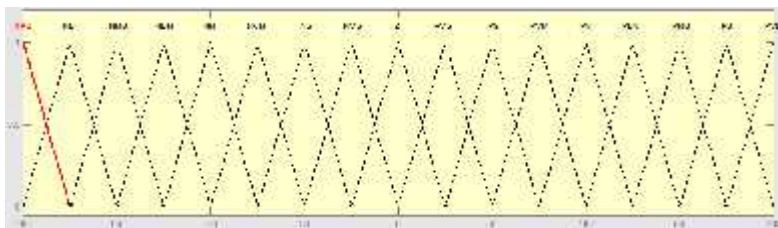
Karena nilai yang diinginkan berjarak dari -200 sampai 200 maka penentuan fungsi keanggotaan (*membership function*) untuk nilai pX dapat dilihat pada Gambar 3.9.



Gambar 3.10 Fungsi keanggotaan nilai pX

• Langkah 2

Sedangkan untuk penentuan fungsi keanggotaan (*membership function*) nilai pY yang berjarak dari -200 sampai 200 dapat dikategorikan pada Gambar 3.10. Fungsi keanggotaan nilai pY tidak ada perbedaan dengan fungsi keanggotaan nilai pX berjumlah 10.



Gambar 3.11 Fungsi keanggotaan nilai pY

• Langkah 3

Selanjutnya adalah menentukan fungsi keanggotaan keluaran dari sistem *fuzzy*. Pada Tabel 3.2 terlihat bahwa jarak keluaran berkisar dari -90 sampai 90. Jarak ini tidak melebihi dari jarak jangkauan untuk θ_1 sesuai dengan DH-Parameter robot manipulator Denso 6-DOF yang mempunyai batasan -170 sampai 170. Dalam Tugas Akhir ini metode sistem *fuzzy* yang dipakai adalah Sugeno, maka dari itu untuk fungsi keanggotaan keluaran mengikuti besarnya setiap nilai yang dihasilkan dari beberapa kemungkinan pada Tabel 3.2.

• Langkah 4

Setelah penentuan fungsi keanggotaan masukan dan keluaran, selanjutnya adalah menentukan Basis Aturan (*Rule Base*) yang menyatakan hubungan masukan dan keluaran tersebut. Dengan memacu dari data pada Tabel 3.2 maka dapat dibuat Basis Aturan sistem *fuzzy*. Dapat dilihat bahwa basis aturan yang berjumlah 81 dikarenakan keluaran metode Sugeno yang berupa *singleton – singleton*, jadi setiap *singleton* mewakili setiap nilai dari beberapa kombinasi fungsi keanggotaan masukan sistem *fuzzy*.

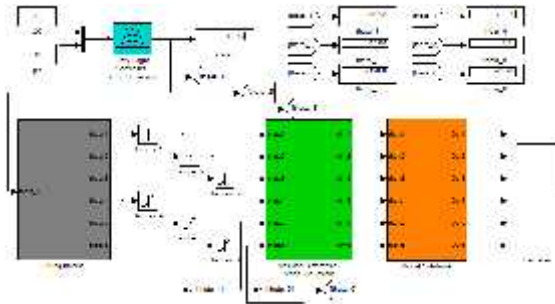
• Langkah 5

Penentuan proses agregasi pada sistem *fuzzy* dengan metode Sugeno adalah *max*, sedangkan untuk proses defuzzifikasinya menggunakan *wtaver* atau *average* pada *Toolbox* yang digunakan.

3.5 Diagram blok keseluruhan

Penggunaan *Simulink* untuk Tugas Akhir ditujukan agar jelasnya aliran (*flow*) proses jalannya rancangan yang dibuat. Fungsi *scope* yang lebih mudah untuk menggambarkan hasil keluaran yang berupa garis. Dalam proses untuk memasukkan nilai yang diinginkan juga akan lebih jelas tata letaknya

Pemodelan untuk keseluruhan perhitungan kinematika terbalik menggunakan *m-file* yang ada pada blok diagram *Simulink*. Pemberian masukan untuk posisi yang dituju robot dapat diberikan pada *m-file* yang ada pada *Simulink*. Pada Gambar 3.11 ditunjukkan keseluruhan blok diagram untuk sistem.



Gambar 3.12 Diagram blok keseluruhan

Dari Gambar 3.11 dapat dilihat bahwa terdapat blok *toolbox fuzzy*, subsistem hitung *inverse*, sub sistem masukan differensial model solidwork dan sub sistem model solidwork. Sub sistem hitung *inverse* berisi *m-file* perhitungan kinematika terbalik dan pemberian masukan posisi yang diinginkan. Sub sistem masukan differensial model solidwork berisi beberapa blok diagram differensial, ini berfungsi untuk menggerakkan sendi robot dibutuhkan masukan derajat, kecepatan derajat dan percepatan derajat. Sub sistem model solidwork berupa blok diagram yang terbentuk dari pemodelan yang telah dibuat dari *software* SOLIDWORK 2014.

BAB IV HASIL DAN SIMULASI

Pada bab ini menjelaskan hasil percobaan simulasi beserta analisisnya. Hasil simulasi ini mengacu pada konsep perhitungan kinematika terbalik (*inverse kinematics*) dan perancangan sistem *fuzzy* yang telah dibahas pada Bab III dan hasil simulasinya menggunakan perangkat lunak MATLAB R2012a.

4.1 Pengujian Pemodelan *Fuzzy* dengan Beberapa Masukan

Sistem *fuzzy* yang telah dirancang untuk menemukan hasil *theta* 1 dengan beberapa variasi masukan nilai pX dan pY dibahas pada subbab ini. Beberapa hasil keluaran dari sistem *fuzzy* dibandingkan dengan keluaran hasil perhitungan terlihat agak berbeda. Hasil dari perhitungan tersebut dapat dilihat pada Tabel 4.1, Tabel 4.2 dan Tabel 4.3, dimana masing-masing tabel telah divariasi nilai masukannya.

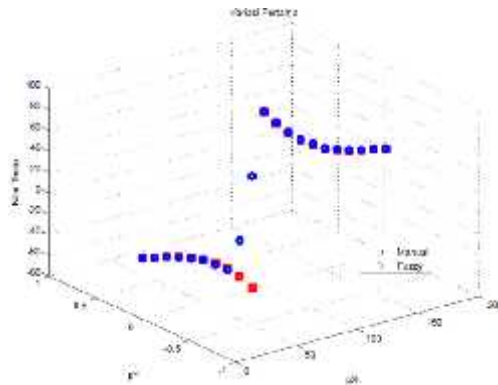
Tabel 4.1 Hasil Variasi hitungan pertama

pX	pY	Manual	Fuzzy	Absolut Error
0	0	-21.38	-21.4	0.02
10	0	-23.55	-23.9	0.35
20	0	-26.17	-26.4	0.23
30	0	-29.4	-29.9	0.5
40	0	-33.44	-34.2	0.76
50	0	-38.56	-38.6	0.04
60	0	-45.17	-46.7	1.53
70	0	-53.72	-54.8	1.08
80	0	-64.64	-29.6	35.04
90	0	-77.92	28.9	106.82
100	0	87.36	87.4	0.04
110	0	72.97	74.5	1.53
120	0	60.48	61.6	1.12
130	0	50.43	51.4	0.97
140	0	42.63	44	1.37
150	0	36.59	36.6	0.01
160	0	31.89	32.6	0.71
170	0	28.18	28.6	0.42

Tabel 4.1 (lanjutan) Variasi hitungan pertama

pX	pY	Manual	Fuzzy	Absolut Error
180	0	25.18	25.4	0.22
190	0	22.73	23.1	0.37
200	0	20.69	20.7	0.01
Rata - rata Error				7.292

Hasil yang didapat menggunakan sistem *fuzzy* jika dibandingkan dengan perhitungan manual terdapat perbedaan. Dari Tabel 4.1 dapat dilihat bahwa variasi pertama menunjukkan perhitungan manual dan *fuzzy* yang berbeda pada tiap pX . Hasil respon yang berbeda pada tiap pX dapat dilihat pada Gambar 4.1. Desain sistem *fuzzy* ditetapkan pada saat pX bernilai 0, 50, 100, 150 atau 200, sehingga memperoleh hasil yang akurat pada titik tersebut. Selain nilai pX tersebut hasil perhitungan manual dan *fuzzy* akurat. Perbedaan yang sangat signifikan terlihat pada masukan $pX = 90$ dan $pY = 0$, dimana nilai dari absolut errornya bernilai 106,82. Perbedaan yang sangat besar ini disebabkan karena



Gambar 4.1 Grafik Variasi pertama *theta 1*

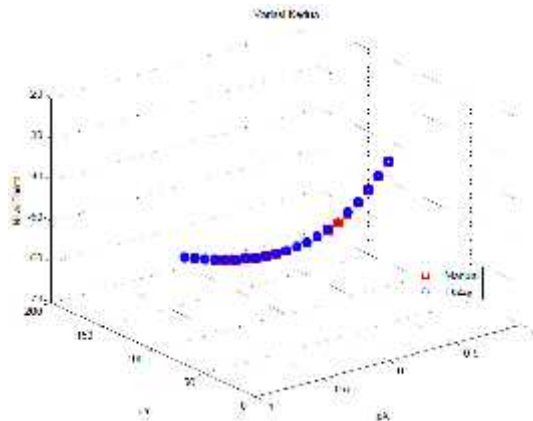
Variasi pertama yang didapat sebenarnya hampir mendekati keluaran yang diinginkan. Tetapi hasil yang mempunyai error besar terjadi saat mendekati kemungkinan ke-6 atau saat nilai $pX=60$ dan $pY=0$, hasil yang didapat mulai jauh dari yang diharapkan. Dengan hasil yang didapat dari variasi pertama ini, maka variasi kedua dilakukan dengan memasukkan nilai $pX=0$ untuk seluruh kemungkinan dan hanya

nilai pY saja yang divariasikan. Beberapa data yang didapatkan dicatat pada Tabel 4.2 untuk perbandingan hasil perhitungan manual dan sistem *fuzzy*.

Tabel 4.2 Hasil Variasi hitungan kedua

pX	pY	Manual	Fuzzy	Absolut Error
0	0	-21.38	-21.4	0.02
0	10	-26.25	-26	0.25
0	20	-30.75	-30.6	0.15
0	30	-34.87	-34.7	0.17
0	40	-38.61	-38.3	0.31
0	50	-42	-42	0
0	60	-45.06	-44.8	0.26
0	70	-47.83	-47.7	0.13
0	80	-50.33	-50.2	0.13
0	90	-52.59	-52.4	0.19
0	100	-54.65	-54.6	0.05
0	110	-56.51	-56.4	0.11
0	120	-58.2	-58.1	0.1
0	130	-59.75	-59.7	0.05
0	140	-61.17	-61.1	0.07
0	150	-62.47	-62.5	0.03
0	160	-63.67	-63.6	0.07
0	170	-64.77	-64.7	0.07
0	180	-65.79	-65.8	0.01
0	190	-66.73	-66.7	0.03
0	200	-67.61	-67.6	0.01
		Rata - rata Error		0.105

Keluaran yang dihasilkan pada variasi kedua menunjukkan keselarasannya hasil perhitungan manual dengan sistem *fuzzy* yang dibuat. Hasil yang didapat ini menunjukkan bahwa saat nilai $pX = 0$ saat berapapun nilai pY yang diberikan, maka hasil yang didapat hampir akurat. Lebih jelasnya dibuatlah plot yang menunjukkan hampir akuratnya data yang dihasilkan, seperti dapat dilihat pada Gambar 4.2.



Gambar 4.2 Grafik Variasi kedua θ_1

Pengujian variasi terakhir yang dilakukan dapat dilihat pada Tabel 4.3, dimana nilai untuk pX dimulai dari 200 sampai 0 sedangkan nilai pY dimulai dari 0 sampai 200. Hal ini dilakukan untuk menguji sistem *fuzzy* yang dibuat, apakah sesuai dengan yang diharapkan atau tidak.

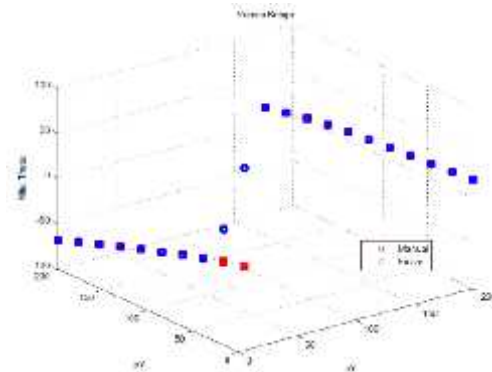
Tabel 4.3 Hasil Variasi hitungan ketiga

pX	pY	Manual	Fuzzy	Absolut Error
200	0	20.69	20.7	0.01
190	10	27.83	27.8	0.03
180	20	35.56	35.5	0.06
170	30	43.64	43.7	0.06
160	40	51.78	51.8	0.02
150	50	59.66	59.7	0.04
140	60	67.01	67	0.01
130	70	73.67	73.7	0.03
120	80	79.58	79.6	0.02
110	90	84.76	84.8	0.04
100	100	89.27	89.3	0.03
90	110	-88.83	21.2	108.03
80	120	-83.44	-47.4	36.04
70	130	-80.49	-80.5	0.01
60	140	-77.91	-77.9	0.01

Tabel 4.3 (lanjutan) Variasi hitungan ketiga

pX	pY	Manual	Fuzzy	Absolut Error
50	150	-75.65	-75.6	0.05
40	160	-73.65	-73.6	0.05
30	170	-71.88	-71.9	0.02
20	180	-70.29	-70.3	0.01
10	190	-68.88	-68.9	0.02
0	200	-67.61	-67.6	0.01
Rata - rata Error				6.886

Dari Tabel 4.3 dapat terlihat bahwa data memiliki Absolut error yang sedikit untuk beberapa kemungkinan yang diinginkan. Keakuratan data yang dikeluarkan diatas dapat lebih jelas dilihat pada Gambar 4.3.



Gambar 4.3 Grafik Variasi ketiga *theta* 1

Gambar 4.3 menjelaskan bahwa keluaran sistem *fuzzy* hanyalah akurat untuk beberapa kemungkinan saja. Beberapa kemungkinan ini sesuai dengan basis aturan yang telah dirancang sebelumnya pada Bab 3. Basis aturan menjelaskan bahwa untuk beberapa kemungkinan nilai yang ditetapkan untuk nilai pX dan pY maka nilai keluaran sesuai dengan nilai dari *singleton*-nya. Misal, jika diberi masukan $pX = 100$ dan $pY = 100$ maka keluaran sama dengan 89,3. Nilai ini akurat karena basis aturan yang telah dirancang memang harus mengeluarkan nilai tersebut. Akan tetapi untuk nilai-nilai yang mendekati masukan tidak dapat sesuai dengan perhitungan manualnya.

4.2 Pengujian Model Rumus *Inverse Kinematics*

Pengujian model rumus kinematika terbalik hanya digunakan satu kali masukan, yaitu dengan memasukkan matriks Persamaan 4.1 dibawah.

$$\begin{bmatrix} -0.0188 & 0.4154 & 0.9095 & 95 \\ 0.4810 & 0.8012 & -0.3560 & 60 \\ -0.8765 & 0.4307 & -0.2148 & -30 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.1)$$

Dari Persamaan 4.1 diketahui bahwa untuk persamaan kinematika terbalik hanya nilai translasi dari nilai pX , pY dan pZ . Maka akan didapatkan beberapa solusi seperti terlihat pada Tabel 4.4.

Tabel 4.4 Bebeapa hasil solusi kinematika terbalik

Ke-	<i>theta1</i>	<i>theta2</i>	<i>theta3</i>	<i>theta4</i>	<i>theta5</i>	<i>theta6</i>
1	-88.12	-57.63	-119	-72.89	80.19	136.2
2	-88.12	-57.63	26.68	-70.37	89.19	-283.3
3	-88.12	-100	-13.71	-76.1	75.97	109.6
4	-88.12	-100	133.1	-70.35	90.05	360
5	90.08	-36.45	-118.2	69.19	85.29	204.1
6	90.08	-36.45	30.46	75.36	102.7	103.2
7	90.14	-100	119.4	88,3	109.4	56.2
8	90.14	-100	-34.2	74.36	74.85	109.4

Saat MATLAB dijalankan, model yang telah dibuat pada SOLIDWORK juga akan mengikuti besarnya perubahan rotasi pada setiap sendi robot. Dari setiap kemungkinan diatas didapat posisi akhir dari robot yang dapat dilihat pada Gambar 4.1. Beberapa kemungkinan ini menunjukkan bahwa solusi yang didapat dari kinematika terbalik adalah unik. Diartikan unik karena untuk menuju titik yang diinginkan dapat dicapai melalui beberapa gerakan atau kemungkinan ini. Bab III yang menjelaskan tentang persamaan kinematika terbalik setiap sendi memiliki beberapa kemungkinan seperti tersebut pada Tabel 4.4. keseluruhan perincian tentang kemungkinan diatas dijelaskan dalam beberapa maksud sebagai berikut:

- **Theta 1**

Mempunyai 2 solusi, pertama sesuai dengan penyelesaian persamaan yang diberikan, kedua penyelesaian yang didapat ditambah 180 derajat agar menghasilkan solusi kedua.

- **Theta 2**

Mempunyai 2 solusi setiap masing – masing besarnya θ_1 , sesuai dengan penyelesaian persamaan yang diberikan akan menghasilkan dua solusi..

- **Theta 3**

Mempunyai 2 solusi setiap masing – masing besarnya θ_1 dan θ_2 , sesuai dengan penyelesaian persamaan yang diberikan akan menghasilkan dua solusi..

- **Theta 4**

Mempunyai 1 solusi setiap masing – masing besarnya θ_1 , θ_2 , θ_3 dan θ_5 , hasil didapat dengan menyelesaikan persamaan yang diberikan.

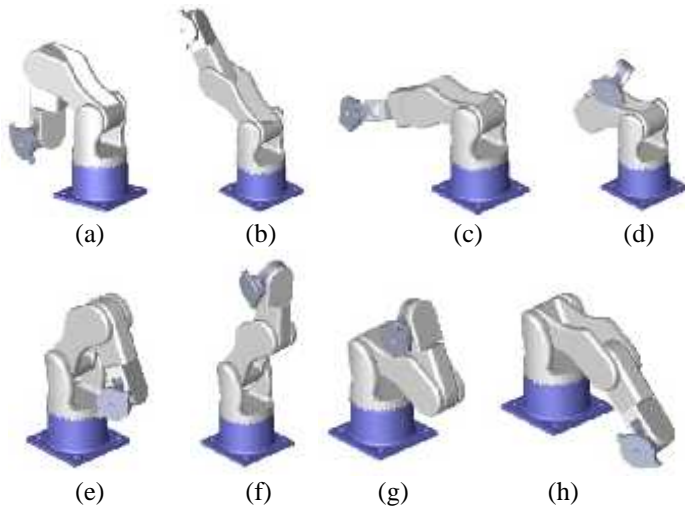
- **Theta 5**

Mempunyai 1 solusi setiap masing – masing besarnya θ_1 , θ_2 , dan θ_3 , hasil didapat dengan menyelesaikan persamaan yang diberikan.

- **Theta 6**

Mempunyai 1 solusi setiap masing – masing besarnya θ_1 , θ_2 , θ_3 dan θ_5 , hasil didapat dengan menyelesaikan persamaan yang diberikan.

Dari Gambar 4.4 terlihat bahwa tidak tepatnya keseluruhan solusi yang dihasilkan, hanya solusi ke 4 dan 7 saja yang sama titik akhirnya. Diperlukannya revisi untuk perhitungan persamaan yang harus diselesaikan kembali.



Gambar 4.4 Beberapa Gerakan Robot solusi kinematika terbalik (a) Solusi 1 (b) Solusi 2 (c) Solusi 3 (d) Solusi 4 (e) Solusi 5 (f) Solusi 6 (g) Solusi 7 (h) Solusi 8

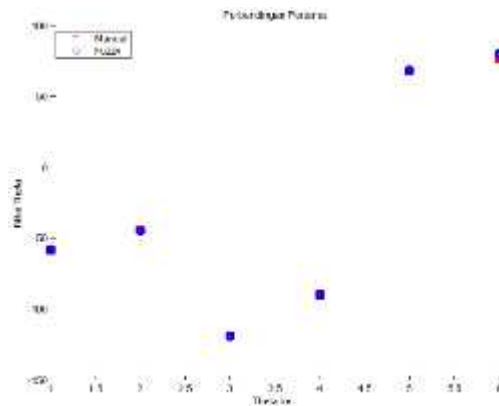
4.3 Perbandingan hasil solusi

Pada sub bab ini akan dibahas mengenai solusi perbandingan yang dihasilkan antara perhitungan manual dengan sistem *fuzzy* yang dibuat. Dengan memberikan beberapa nilai masukan titik pX , pY dan pZ yang berbeda pada setiap perbandingan. Perbandingan pertama dimasukkan nilai $pX = 55$, $pY = 30$ dan $pZ = -10$, hasil yang didapatkan terlihat pada Tabel 4.5.

Tabel 4.5 Hasil Perbandingan pertama

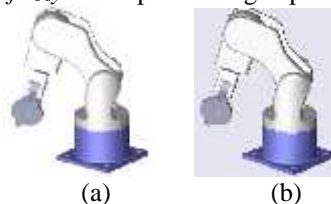
$pX = 55$	$pY = 30$	$pZ = -10$	Manual	Fuzzy	Absolut Error
Theta 1			-57.73	-57.78	0.05
Theta 2			-43.98	-44.79	0.81
Theta 3			-119	-119	0
Theta 4			-90	-90	0
Theta 5			67.91	68.82	0.91
Theta 6			76.91	80.49	3.58
			Rata - rata Error		0.8917

Dari hasil yang didapatkan untuk setiap besarnya θ terlihat bahwa setiap hasil sistem *fuzzy* hampirlah akurat jika dibandingkan dengan hasil perhitungan manual. Pada θ 3 nilai yang didapat adalah -119, karena sesuai dengan blok diagram keseluruhan pada Bab III yang membatasi pergerakan sendi ke 3 sampai nilai paling bawah yaitu -119, kondisi ini juga sesuai dengan batas pada sendi ke 3 yang ada pada DH-Parameter. Hanya nilai θ 6 saja yang kurang mendekati hasil perhitungan manual. Lebih jelasnya tentang beberapa nilai pada Tabel 4.5 terlihat dengan grafik pada Gambar 4.5.



Gambar 4.5 Grafik Perbandingan Pertama

Kurang tepatnya hasil untuk sendi terakhir atau biasa disebut *end-effector* dan sendi pertama penghubung *base frame* dengan robot dapat diperlihatkan pergerakannya melalui Gambar 4.6. Hasil untuk posisi robot pun tidak begitu terlalu jauh dari yang diharapkan bila menggunakan sistem *fuzzy* untuk perbandingan pertama ini.



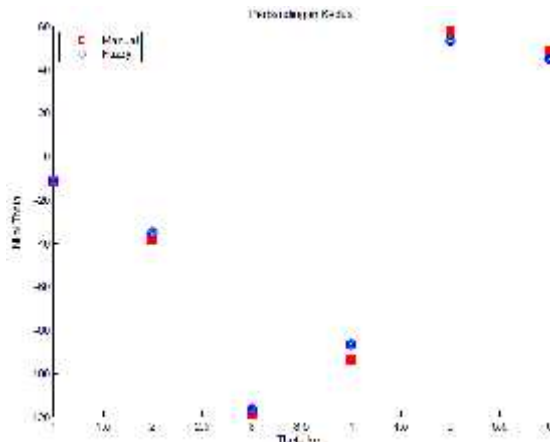
Gambar 4.6 Gerakan Robot Perbandingan Pertama (a) Manual (b) Fuzzy

Perbandingan kedua dilakukan dengan memasukan nilai $pX = 30$, $pY = -25$ dan $pZ = 5$, secara lebih jelasnya dapat dilihat pada Tabel 4.6.

Tabel 4.6 Hasil Perbandingan kedua

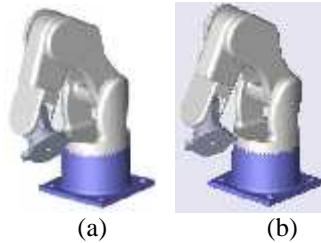
$pX = 30$	$pY = -25$	$pZ = 5$	Manual	Fuzzy	Absolut Error
Theta 1			-11.6	-11	0.6
Theta 2			-37.93	-35.05	2.88
Theta 3			-119	-116.4	2.6
Theta 4			-93.6	-86.54	7.06
Theta 5			57.76	53.36	4.4
Theta 6			48.9	45	3.9
			Rata-rata Error		3.573

Hasil dari seluruh solusi pada Tabel 4.6 menunjukkan data akurat yang mempunyai rata-rata error sebesar 3,573. Perbandingan kedua perhitungan ini hampir menyamai hasil dari perbandingan pertama, karena mempunyai kesamaan kondisi untuk *theta* 6 yang kurang mendekati hasil dari perhitungan manual. Grafik yang dihasilkan menurut hasil perbandingan kedua dapat dilihat pada Gambar 4.7.



Gambar 4.7 Grafik Perbandingan Kedua

Simulasi pergerakan robot Denso 6-DoF untuk perbandingan kedua ini dapat dilihat pada Gambar 4.8.



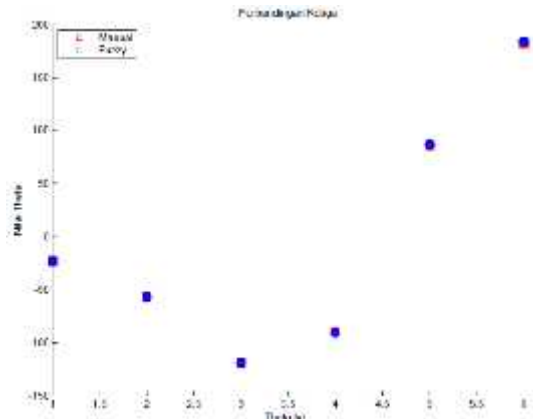
Gambar 4.8 Gerakan Robot Perbandingan Kedua (a) Manual (b) Fuzzy

Pengujian perbandingan yang terakhir dilakukan dengan memasukkan nilai $pX = -60$, $pY = 30$ dan $pZ = 8$. Tabel 4.7 merepresentasikan hasil setiap θ untuk perbandingan perhitungan manual dan sistem *fuzzy* yang dibuat.

Tabel 4.7 Hasil Perbandingan ketiga

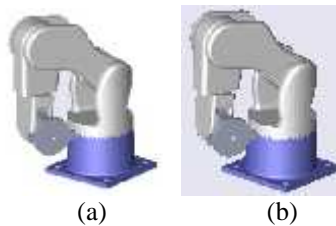
$pX = -60$	$pY = 30$	$pZ = 8$	Manual	Fuzzy	Absolut Error
Theta 1			-23.39	-23.67	0.28
Theta 2			-56.44	-57.03	0.59
Theta 3			-119	-119	0
Theta 4			-90	-90.94	0.94
Theta 5			85.7	86.59	0.89
Theta 6			181.8	183.7	1.9
			Rata-rata Error		0.767

Berbeda dengan perbandingan pertama dan kedua, pada perbandingan ketiga ini perbedaan hasil yang didapat hampir semuanya akurat. Agar lebih jelasnya dapat dilihat grafiknya pada Gambar 4.9.



Gambar 4.9 Grafik Perbandingan Ketiga

Sesuai dengan hasil pada Gambar 4.9 bahwa keluaran sistem *fuzzy* dapat mencapai perhitungan manual yang diinginkan dengan rata-rata error sebesar 0.767. Dengan didapatkannya hasil bahwa *theta 1* saja yang sedikit errornya, maka pergerakan robot untuk perbandingan ketiga ini hampir sama untuk perhitungan manual dan sistem *fuzzy*-nya. Gambar 4.10 menunjukkan pergerakan robot antara perhitungan manual dan sistem *fuzzy*.



Gambar 4.10 Gerakan Robot Perbandingan Ketiga (a) Manual (b) Fuzzy

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil simulasi, ada beberapa hal yang dapat disimpulkan dari penelitian pada Tugas Akhir ini. Pertama, beberapa percobaan keluaran hasil sistem *fuzzy* dengan perhitungan manual untuk perhitungan nilai θ 1 mempunyai rata-rata error sebesar 4.761 derajat. Kedua, untuk beberapa percobaan dalam menentukan hasil pergerakan robot dengan mencari nilai seluruh θ dengan membandingkan antara keluaran sistem *fuzzy* dengan perhitungan manual mempunyai rata-rata error sebesar 1.744 derajat. Ketiga, Perancangan untuk pencarian persamaan θ dapat dicari tanpa melakukan *inverse* untuk matriks Transformasi Homogen.

5.2 Saran

Sebagai pengembangan penelitian, pertama penulis menyarankan untuk pemodelan sistem *fuzzy* haruslah akurat dan sesuai dengan keluaran perhitungan manual. Dalam pengklasifikasian fungsi keanggotaan masukan serta basis aturan haruslah memiliki banyak aturan sehingga mempunyai hasil yang tepat seperti yang telah dibahas oleh penulis.

Kedua, pengembangan simulasi selanjutnya, dengan menghitung dinamika robot Denso 6-DoF. Ketiga, perhitungan kinematika terbalik dilakukan dua kali, dengan metode aljabar dan geometri agar terklarifikasinya persamaan kinematika terbalik yang didapat.

--- *Halaman ini sengaja dikosongkan* ---

LAMPIRAN A

A.1 Penurunan Model *Inverse Kinematics* Robot Denso 6-DoF

Dengan memanfaatkan rumus untuk perhitungan masing-masing θ yang terbentuk pada Persamaan A.1. Nilai masing-masing matriks telah disebutkan sebelumnya pada Persamaan 3.1 sampai 3.6.

$${}^3T_4 {}^4T_5 {}^5T_6 = {}^2T_3 {}^1T_2 {}^0T_1 {}^{0T}_6 \quad (A.1)$$

Maka dapat dipisahkan seperti berikut:

$${}^3T_4 {}^4T_5 {}^5T_6 = \begin{bmatrix} R_4 R_5 R_6 & R_4 R_5 p_6 + R_4 p_5 + p_4 \\ 0 & 1 \end{bmatrix} \quad (A.2)$$

$${}^2T_3 {}^1T_2 {}^0T_1 {}^{0T}_6 = \begin{bmatrix} R_3^T R_2^T R_1^T R & R_3^T R_2^T R_1^T p - R_3^T R_2^T R_1^T p_1 - R_3^T R_2^T p_2 - R_3^T p_3 \\ 0 & 1 \end{bmatrix} \quad (A.3)$$

Maka dapat dibentuk relasinya seperti berikut:

$$R_4 R_5 R_6 = R_3^T R_2^T R_1^T R \quad (A.4)$$

$$R_4 R_5 p_6 + R_4 p_5 + p_4 = R_3^T R_2^T R_1^T p - R_3^T R_2^T R_1^T p_1 - R_3^T R_2^T p_2 - R_3^T p_3 \quad (A.5)$$

Dimana Relasi A.5 persamaan didapat

$$\begin{bmatrix} -d_6 c_4 s_5 \\ -d_6 s_4 s_5 \\ d_4 + d_6 c_5 \end{bmatrix} = \begin{bmatrix} s_{23}(p_z - d_1 - a_2 s_2) + c_{23}(-a_2 c_2 + p_x c_1 + p_y s_1 - a_1) - a_3 \\ p_y c_1 - p_x s_1 \\ c_{23}(p_z - d_1 - a_2 s_2) + s_{23}(a_2 c_2 - p_x c_1 - p_y s_1 + a_1) \end{bmatrix} \quad (A.6)$$

Persamaan A.6 merupakan penyederhanaan dari

$$\begin{bmatrix} p_z(c_2 s_3 + c_3 s_2) - d_1(c_2 s_3 + c_3 s_2) - a_3 c_3^2 - a_3 s_3^2 - a_2 c_2(c_2 c_3 - s_2 s_3) + p_x c_1(c_2 c_3 - s_2 s_3) \\ -a_2 s_2(c_2 s_3 + c_3 s_2) + p_y s_1(c_2 c_3 - s_2 s_3) - a_1 c_1^2(c_2 c_3 - s_2 s_3) - a_1 s_1^2(c_2 c_3 - s_2 s_3) \\ p_y c_1 - p_x s_1 \\ p_z(c_2 c_3 + s_2 s_3) - d_1(c_2 c_3 + s_2 s_3) + a_2 c_2(c_2 s_3 - s_2 c_3) - p_x c_1(c_2 s_3 - s_2 c_3) \\ -a_2 s_2(c_2 c_3 + s_2 s_3) - p_y s_1(c_2 s_3 - s_2 c_3) + a_1 c_1^2(c_2 s_3 - s_2 c_3) + a_1 s_1^2(c_2 s_3 - s_2 c_3) \end{bmatrix}$$

Kemudian dari Persamaan A.4 persamaan didapat:

Ruas Kiri:

$$\begin{bmatrix} s_{23}n_z + c_{23}(n_x c_1 + n_y s_1) & s_{23}o_z + c_{23}(o_x c_1 + o_y s_1) & s_{23}a_z + c_{23}(a_x c_1 + a_y s_1) \\ n_y c_1 - n_x s_1 & o_y c_1 - o_x s_1 & a_y c_1 - a_x s_1 \\ c_{23}n_z + s_{23}(-n_x c_1 - n_y s_1) & c_{23}o_z + s_{23}(-o_x c_1 - o_y s_1) & c_{23}a_z + s_{23}(-a_x c_1 - a_y s_1) \end{bmatrix} \quad (\text{A.7})$$

Ruas Kanan:

$$\begin{bmatrix} c_4 c_5 c_6 - s_4 s_6 & -c_6 s_4 - c_4 c_5 s_6 & -c_4 s_5 \\ c_4 s_6 + c_5 c_6 s_4 & c_4 c_6 - c_5 s_4 s_6 & -s_4 s_5 \\ c_6 s_5 & -s_5 s_6 & c_5 \end{bmatrix} \quad (\text{A.8})$$

Sehingga bisa didapatkan beberapa Persamaan seperti yang telah disebutkan pada Bab 3 untuk persamaan 3.29 sampai 3.35.

$$a_3 - d_6 c_4 s_5 = s_{23}(p_z - d_1 - a_2 s_2) + c_{23}(-a_2 c_2 + p_x c_1 + p_y s_1 - a_1) \quad (\text{A.9})$$

$$-d_6 s_4 s_5 = p_y c_1 - p_x s_1 \quad (\text{A.10})$$

$$d_4 + d_6 c_5 = c_{23}(p_z - d_1 - a_2 s_2) + s_{23}(a_2 c_2 - p_x c_1 - p_y s_1 + a_1) \quad (\text{A.11})$$

$$c_{23}n_z + s_{23}(-n_x c_1 - n_y s_1) = c_6 s_5 \quad (\text{A.12})$$

$$c_{23}o_z + s_{23}(-o_x c_1 - o_y s_1) = -s_5 s_6 \quad (\text{A.13})$$

$$c_{23}a_z + s_{23}(-a_x c_1 - a_y s_1) = c_5 \quad (\text{A.14})$$

$$a_y c_1 - a_x s_1 = -s_4 s_5 \quad (\text{A.15})$$

$$s_{23}a_z + c_{23}(a_x c_1 + a_y s_1) = -c_4 s_5 \quad (\text{A.16})$$

Sehingga bisa dicari untuk persamaan setiap *theta* yang dicari

- Mencari θ_1

Dari persamaan (A.10) dan (A.15) maka akan dapat dihitung:

$$\frac{p_y c_1 - p_x s_1}{d_6} = -s_4 s_5$$

$$a_y c_1 - a_x s_1 = -s_4 s_5$$

$$\left(\frac{p_y}{d_6} - a_y \right) c_1 + \left(\frac{-p_x}{d_6} + a_x \right) s_1 = 0 \quad (\text{A.17})$$

Maka dengan memanfaatkan solusi dari persamaan (10) didapat

$$\theta_1 = A \tan 2 \left(- \left(\frac{p_y}{d_6} - a_y \right), \left(\frac{-p_x}{d_6} + a_x \right) \right)$$

Atau

$$\theta_1 = A \tan 2 \left(\left(\frac{p_y}{d_6} - a_y \right), - \left(\frac{-p_x}{d_6} + a_x \right) \right)$$

- Mencari θ_2

Dengan mengkuadratkan persamaan (A.14) dan (A.16) kemudian menjumlahkannya akan didapat persamaan baru, seperti yang terjabar dibawah ini

$$\begin{aligned} (c_{23}a_z)^2 - (s_{23}(a_xc_1 + a_ys_1))^2 &= (c_5)^2 \\ (s_{23}a_z)^2 + (c_{23}(a_xc_1 + a_ys_1))^2 &= (-c_4c_5)^2 \\ \hline (a_z)^2 + (a_xc_1 + a_ys_1)^2 &= (-c_4c_5)^2 + (c_5)^2 \end{aligned} \quad (\text{A.18})$$

Karena dari penjabaran $(a_xc_1 + a_ys_1)^2$ akan didapatkan

$$(a_xc_1)^2 + 2a_xa_ys_1c_1 + (a_ys_1)^2$$

Kemudian kuadratkan persamaan (1) dan (3) lalu menjumlahkannya akan didapat persamaan baru, seperti yang terjabar dibawah ini

$$\begin{aligned} (c_{23}(a_2c_2 - c_1p_x - s_1p_y + a_1))^2 - (s_{23}(p_z - d_1 - a_2s_2))^2 - (a_3)^2 &= (-d_6c_4c_5)^2 \\ (s_{23}(a_2c_2 - c_1p_x - s_1p_y + a_1))^2 + (c_{23}(p_z - d_1 - a_2s_2))^2 - (d_4)^2 &= (d_6c_5)^2 \\ \hline \end{aligned}$$

$$(a_2c_2 - c_1p_x - s_1p_y + a_1)^2 - (p_z - d_1 - a_2s_2)^2 - (a_3)^2 - (d_4)^2 = (-d_6c_4c_5)^2 + (d_6c_5)^2 \quad (\text{A.19})$$

Kemudian nilai ini $(p_z - d_1 - a_2s_2)^2$ dijabarkan untuk mendapatkan persamaan lebih mudah seperti terlihat pada Persamaan A.20.

$$(p_z)^2 + (d_1)^2 + (a_2s_2)^2 - 2d_1p_z - 2a_2s_2p_z + 2d_1a_2s_2 \\ 2a_2((d_1 - p_z)s_2) + (d_1 - p_z)^2 + (a_2s_2)^2 \quad (\text{A.20})$$

Kemudian nilai dari persamaan $(a_2c_2 - c_1p_x - s_1p_y + a_1)^2$ bila dijabarkan akan mendapatkan persamaan yang lebih mudah seperti terlihat

$$(a_2c_2)^2 + (c_1p_x)^2 + (s_1p_y)^2 + (a_1)^2 - 2a_2c_2c_1p_x - 2a_2c_2s_1p_y - 2a_1s_1p_y \\ 2a_2((-c_1p_x - s_1p_y + a_1)c_2) + (-c_1p_x - s_1p_y + a_1)^2 + (a_2c_2)^2 \quad (\text{A.21})$$

Kemudian dengan menggabungkan persamaan (A.20) dan (A.21) maka akan didapatkan

$$2a_2[(d_1 - p_z)s_2 + (-c_1p_x - s_1p_y + a_1)c_2] + (d_1 - p_z)^2 + (-c_1p_x - s_1p_y + a_1)^2 + (a_2)^2 \quad (\text{A.22})$$

$$\text{Karena nilai dari } (a_2c_2)^2 + (a_2s_2)^2 = (a_2)^2$$

Kemudian berdasarkan penjabaran tersebut dikembalikan kembali nilai untuk Persamaan A.22 ke Persamaan A.19 maka akan didapatkan persamaan:

$$2a_2[(d_1 - p_z)s_2 + (-c_1p_x - s_1p_y + a_1)c_2] + (d_1 - p_z)^2 + (-c_1p_x - s_1p_y + a_1)^2 + (a_2)^2 - (a_3)^2 - (d_4)^2 = (-d_6c_4c_5)^2 + (d_6c_5)^2 \quad (\text{A.23})$$

Dan disederhanakan untuk menjadi

$$\frac{2a_2[(d_1 - p_z)s_2 + (-c_1p_x - s_1p_y + a_1)c_2] + (d_1 - p_z)^2 + (-c_1p_x - s_1p_y + a_1)^2 + (a_2)^2 - (a_3)^2 - (d_4)^2}{d_6} = (-c_4c_5)^2 + (c_5)^2 \quad (\text{A.24})$$

Persamaan (A.24) tersebut dikurangi dengan persamaan (A.18) yang akan menjadi

$$\frac{2a_2 \left[(d_1 - p_z) s_z + (-c_1 p_x - s_1 p_y + a_1) c_2 \right] + (d_1 - p_z)^2 + (-c_1 p_x - s_1 p_y + a_1)^2 + (a_2)^2 - (a_3)^2 - (d_4)^2}{d_6} - (a_z)^2 - (a_x c_1 + a_y s_1)^2 = 0 \quad (\text{A.25})$$

Lalu dipindah ruaskan agar ruas sebelah kanan agar tidak sama dengan 0 dan disederhanakan agar dapat memenuhi persamaan (9) untuk mencari a dan b

$$\frac{(d_1 - p_z) s_z}{d_6} + \frac{(-c_1 p_x - s_1 p_y + a_1) c_2}{d_6} - \frac{(d_1 - p_z)^2}{2a_2 \times d_6} + \frac{(-c_1 p_x - s_1 p_y + a_1)^2}{2a_2 \times d_6} + \frac{(a_2)^2}{2a_2 \times d_6} + \frac{(a_3)^2}{2a_2 \times d_6} + \frac{(d_4)^2}{2a_2 \times d_6} + \frac{(a_z)^2}{2a_2} + \frac{(a_x c_1 + a_y s_1)^2}{2a_2} \quad (\text{A.26})$$

Dengan asumsi bahwa

$$a = \frac{(d_1 - p_z)}{d_6}$$

$$b = \frac{(-c_1 p_x - s_1 p_y + a_1)}{d_6}$$

$$c = \frac{(d_1 - p_z)^2}{2a_2 \times d_6} + \frac{(-c_1 p_x - s_1 p_y + a_1)^2}{2a_2 \times d_6} + \frac{(a_2)^2}{2a_2 \times d_6} + \frac{(a_3)^2}{2a_2 \times d_6} + \frac{(d_4)^2}{2a_2 \times d_6} + \frac{(a_z)^2}{2a_2} + \frac{(a_x c_1 + a_y s_1)^2}{2a_2}$$

Setelah didapat persamaan diatas maka akan mudah untuk mencari a dan b

- Mencari a dan b
Terlebih dahulu menyederhanakan persamaan (A.9) dengan mengkalikan persamaan tersebut dengan d_6 agar menjadi

$$s_{23} \left(\frac{(p_z - d_1 - a_2 s_2)}{d_6} \right) + c_{23} \left(\frac{(-a_2 c_2 + p_x c_1 + p_y s_1 - a_1)}{d_6} \right) = \frac{a_3}{d_6} - c_4 s_5$$

Lalu persamaan (A.9) yang telah disederhanakan tersebut dikurangi dengan persamaan (A.16) agar menjadi

$$s_{23} \left(\frac{(p_z - d_1 - a_2 s_2)}{d_6} - a_z \right) + c_{23} \left(\frac{(-a_2 c_2 + p_x c_1 + p_y s_1 - a_1)}{d_6} - a_x c_1 - a_y s_1 \right) = \frac{a_3}{d_6}$$

Karena

$$s_{23} = \sin(\theta_2 + \theta_3) = (\sin \theta_2 \cos \theta_3 + \cos \theta_2 \sin \theta_3)$$

$$c_{23} = \cos(\theta_2 + \theta_3) = (\cos \theta_2 \cos \theta_3 - \sin \theta_2 \sin \theta_3)$$

Maka dapat dihitung untuk mencari θ_3 dengan penjabaran sebagai berikut

$$(\sin \theta_2 \cos \theta_3 + \cos \theta_2 \sin \theta_3) \left(\frac{(p_z - d_1 - a_2 s_2)}{d_6} - a_z \right) + (\cos \theta_2 \cos \theta_3 - \sin \theta_2 \sin \theta_3) \left(\frac{(-a_2 c_2 + p_x c_1 + p_y s_1 - a_1)}{d_6} - a_x c_1 - a_y s_1 \right) = \frac{a_3}{d_6} \quad (\text{A.27})$$

Dengan asumsi bahwa

$$a = \left(\frac{(p_z - d_1 - a_2 s_2)}{d_6} - a_z \right)$$

$$b = \left(\frac{(-a_2 c_2 + p_x c_1 + p_y s_1 - a_1)}{d_6} - a_x c_1 - a_y s_1 \right)$$

$$c = \frac{a_3}{d_6}$$

Maka dapat disederhanakan persamaannya sebagai berikut

$$\sin \theta_3 (\cos \theta_2 (a) - \sin \theta_2 (b)) + \cos \theta_3 (\sin \theta_2 (a) - \cos \theta_2 (b)) = c \quad (\text{A.28})$$

Dari Persamaan A.28 tersebut akan didapatkan nilai θ_3

- Mencari θ_5

Dari persamaan (A.14) yang tertulis

$$c_{23} a_z + s_{23} (-a_x c_1 - a_y s_1) = c_5$$

Maka akan didapat persamaan untuk mencari s_5

- Mencari s_4

Dari persamaan (A.15) yang tertulis

$$a_y c_1 - a_x s_1 = -s_4 s_5$$

Dan disederhanakan menjadi

$$-\frac{a_y c_1 - a_x s_1}{s_5} = s_4$$

Maka akan didapat persamaan untuk mencari s_4

- Mencari s_6

Dari persamaan (A.12) yang tertulis

$$c_{23} n_z + s_{23} (-n_x c_1 - n_y s_1) = c_6 c_5$$

Dan disederhanakan menjadi

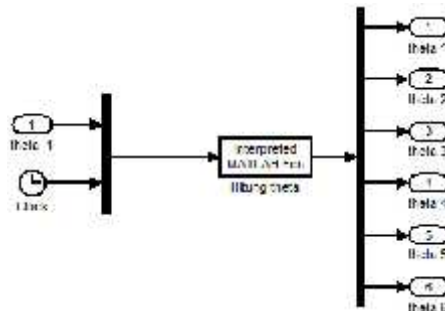
$$\frac{c_{23} n_z + s_{23} (-n_x c_1 - n_y s_1)}{c_5} = c_6$$

Maka akan didapat persamaan untuk mencari s_6

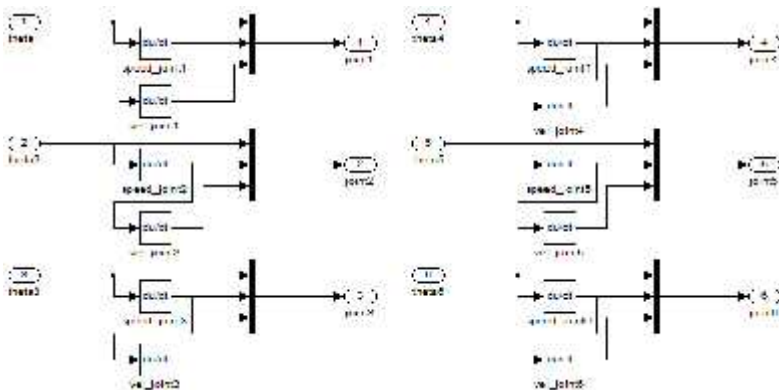
LAMPIRAN B

B.1 Diagram Blok Simulasi pada Simulink MATLAB

Fokus pada Tugas Akhir ini adalah kontrol stabilitas, sehingga model *plant* yang digunakan hanya subsistem stabilitas seperti ditunjukkan pada Gambar B.1 sedangkan untuk subsistem pergerakan *yaw* tidak dicantumkan. Subblok sistem stabilitas diperjelas pada Gambar B.2-B6



Gambar B.1 Subsistem Hitung_Inverse



Gambar B.2 Subsistem Masukan Differensial Model Solidwork

B.2 Listing Program

```
function xo=hitung_theta_fix(xin)

% Deklarasi beberapa variabel
global theta_1 theta_2 theta_3 theta_4 theta_5
theta_6 nX nY nZ oX oY oZ aX aY aZ pX pY pZ tt
global a1 a2 a3 d1 d4 d6 theta_satu
global derajat1 derajat2 derajat3 derajat4
derajat5 derajat6
global jarak1 jarak2 jarak3 jarak4 jarak5 jarak6
global out1 out2 out3 out4 out5 out6 bagi_t
global mpk mqk mrk msk mtk muk to t range jarak
derajat
syms x1 x2 x3 x4 x5 x6

% DH-Paramater robot
a1 = 75;
a2 = 270;
a3 = 90;
d1 = 335;
d4 = 295;
d6 = 108;

to = xin(2);          % Masukan waktu refensi
                       % pergerakan robot

% Posisi akhir dari robot
matriks = [-0.0188 0.4154 0.9095 30
            0.481 0.8012 -0.356 -25
            -0.8765 0.4307 -0.2148 5];

% Pengisian variabel berdasarkan posisi akhir
robot
nX = matriks(1,1);
nY = matriks(2,1);
nZ = matriks(3,1);
oX = matriks(1,2);
oY = matriks(2,2);
```

```

oZ = matriks(3,2);
aX = matriks(1,3);
aY = matriks(2,3);
aZ = matriks(3,3);
pX = matriks(1,4);
pY = matriks(2,4);
pZ = matriks(3,4);

% Jika masukan waktu awal = 0 atau program
  berjalan pertama kali
if to == 0
    theta_satu = xin(1); % Nilai theta 1 dari
                        sistem fuzzy

    % Posisi 0 robot
    out1 = 0;
    out2 = 0;
    out3 = 0;
    out4 = 0;
    out5 = 0;
    out6 = 0;

    bagi_t = 51.463; % perhitungan setiap
                    kenaikan waktu

    % Rumus mencari theta 1
    mpk = solve((pY/d6-aY)*cos(x1)+(-
pX/d6+aX)*sin(x1));

    theta_1 = degtorad(theta_satu/1 % theta dari
                                fuzzy

    % Rumus mencari theta 2
    mqk = solve((d1-pZ)*sin(x2)-
(cos(theta_1)*pX+sin(theta_1)*pY-
(a1))*cos(x2)-(a3^2+d4^2-a2^2-
(cos(theta_1)*pX+sin(theta_1)*pY-(a1))^2-(d1-
pZ)^2+d6*(aZ^2+(aX*cos(theta_1)+(aY)*sin(thet
a_1))^2))/(2*a2));

    theta_2 = mqk(1,1);

```

```

% Rumus mencari theta 3
mrk = solve(sin(x3)*(cos(theta_2)*(((pZ-d1-
a2*sin(theta_2))/d6)-(aZ))-sin(theta_2)*(((
-a2*cos(theta_2)+pX*cos(theta_1)+pY*sin(theta_
1)-(a1))/d6)-aX*cos(theta_1)-
(aY)*sin(theta_1)))+cos(x3)*(sin(theta_2)*(((
pZ-d1-a2*sin(theta_2))/d6)-(aZ)))-
cos(theta_2)*(((
-a2*cos(theta_2)+pX*cos(theta_1)+pY*sin(theta_
1)-(a1))/d6)-aX*cos(theta_1)-
(aY)*sin(theta_1)))-(a3/d6));

theta_3 = mrk(1,1);

% Rumus mencari theta 5
mtk = solve(cos(theta_2+theta_3)*(aZ)-
sin(theta_2+theta_3)*(aX*cos(theta_1)+(aY)*si
n(theta_1))-cos(x5));

theta_5 = mtk;

% Rumus mencari theta 4
msk = solve(cos(theta_1)-
0.9095*sin(theta_1)+sin(x4)*sin(theta_5));

theta_4 = msk;

% Rumus mencari theta 6
muk = solve(cos(theta_2+theta_3)*nZ-
sin(theta_2+theta_3)*(nX*cos(theta_1)+nY*sin(
theta_1))-cos(x6)*cos(theta_5));

theta_6 = imag(muk); % Imajiner theta 6

% Konversi radian ke derajat
derajat1 = radtodeg(theta_1)
derajat2 = radtodeg(theta_2)
derajat3 = radtodeg(theta_3)
derajat4 = radtodeg(theta_4)

```

```

derajat5 = radtodeg(theta_5)
derajat6 = radtodeg(theta_6)

    % Kenaikan berapa derajat setiap waktu
    jarak1 = derajat1/bagi_t; % theta dari fuzzy
    jarak2 = eval(derajat2)/bagi_t;
    jarak3 = eval(derajat3)/bagi_t;
    jarak4 = eval(derajat4)/bagi_t;
    jarak5 = eval(derajat5)/bagi_t;
    jarak6 = eval(derajat6)/bagi_t;

end

% Jika setiap kenaikan waktu maka
if to > t
    out1 = out1 + jarak1;
    out2 = out2 + jarak2;
    out3 = out3 + jarak3;
    out4 = out4 + jarak4;
    out5 = out5 + jarak5;
    out6 = out6 + jarak6;
end

t = to;      % Nilai waktu sebelumnya

% Pergerakan robot berdasarkan besarnya nilai
    setiap theta
xo(1) = out1;
xo(2) = out2;
xo(3) = out3;
xo(4) = out4;
xo(5) = out5;
xo(6) = out6;

```

--- Halaman ini sengaja dikosongkan ---

DAFTAR PUSTAKA

- [1] Liu H., Zhou W., Lai X., Zhu S., “An Efficient Inverse Kinematic Algorithm for a PUMA560-Structured Robot Manipulator”, *International Journal of Advanced Robotic Systems*, vol.10, 236:2013, 18 Maret, 2013.
- [2] Crenganis M., Breaz R., Racs G., Bologa O., “The Inverse Kinematics Solutions of a 7 DOF Robotic Arm Using Fuzzy Logic”, *2012 7th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pp.518-523, 25 October, 2011.
- [3] Denavit J. dan Hartenberg R.S., “A kinematic notation for lower-pair mechanisms based on matrices”, *Journal of Applied Mechanics*, vol. 1, pp.215-221, Juni, 1955.
- [4] Kucuk S., Bingul Z., “Robot Kinematics : Forward and Inverse Kinematics”, *Industrial Robotics : Theory, Modelling and Control*, Intech, Desember, 2006.
- [5] Spong M.W., Vidyasagar M., “*Robot Dynamics and Control*”, New York: Wiley, 1989.
- [6] Hanh L.D., Lin C-Y., “Combining Stereo Vision and Fuzzy Image Based Visual Servoing for Atutonomous Object Grasping Using a 6-DOF Manipulator”, *2012 IEEE International Conference on Robotics and Biomimetics*, pp.1703-1708, 11-14 Desember, 2012
- [7] Naba A., “*Belajar Cepat Fuzzy Logic menggunakan MATLAB*”, Yogyakarta: ANDI, 2009.
- [8] Takagi T. dan Sugeno M., “Fuzzy Identification of Systems and Its Applications to Modeling and Control”, *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-15, No.1, January/February, 1985.
- [9] Quanser / Denso Speciality Plants., “Open-Architecture Enabled with QUARC: Quanser 6-Axis Articulated Robot”, *Denso 6-Axis Open-Architecture Manual*, Document Number: 802, pp.41, Revision: 1.0.

--- *Halaman ini sengaja dikosongkan* ---

RIWAYAT PENULIS



Septian Surya Pradana, lahir di Madiun – Jawa Timur pada tanggal 4 September 1990. Anak pertama dari pasangan Supriyono dan Purwati. Setelah lulus dari SMA Martia Bhakti Bekasi tahun 2009, penulis melanjutkan studi di Diploma 3 Elektronika dan Instrumentasi Universitas Gadjah Mada (UGM) Yogyakarta dan lulus tahun 2012. Kemudian melanjutkan kuliah dengan mengambil program Sarjana Lintas Jalur di Institut Teknologi Sepuluh Nopember Surabaya dengan mengambil Bidang Studi Teknik Sistem Pengaturan, Jurusan Teknik Elektro. Pada bulan Januari 2015, penulis mengikuti seminar dan ujian tugas akhir di Bidang Studi Teknik Sistem Pengaturan, Jurusan Teknik Elektro, ITS Surabaya sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Teknik Elektro.

---Halaman ini sengaja dikosongkan---